VIRTUAL MOUSE MENGGUNAKAN *HAND GESTURE RECOGNITION* BERBASIS *HAND LANDMARK MODEL* UNTUK *POINTING DEVICE*

SKRIPSI

Di Susun Oleh :

JEFFRI DIAN ASMORO

H76217060

PROGRAM STUDI SISTEM INFORMASI

FAKULTAS SAINS DAN TEKNOLOGI

UNIVERSITAS ISLAM NEGERI SUNAN AMPEL

SURABAYA

2022

LEMBAR PERNYATAAN KEASLIAN

Saya yang bertanda tangan dibawah ini,

Nama	: Jeffri Dian Asmoro
NIM	: H76217060
Program Studi	: Sistem Informasi
Angkatan	: 2017

Menyatakan bahwa saya tidak melakukan plagiasi dalam penulisan skripsi saya yang berjudul: " *VIRTUAL MOUSE* MENGGUNAKAN *HAND GESTURE RECOGNITION* BERBASIS *HAND LANDMARK MODEL* UNTUK *POINTING DEVICE* ", apabila suatu saat nanti terbukti saya melakukan Tindakan plagiat, maka saya bersedia menerima sanksi yang telah ditetapkan.

Demikian pernyataan keaslian ini saya buat dengan sebenar-benarnya.

Surabaya, 13 Januari 2022

Yang Menyatakan,

Jeffri Dian Asmoro

NIM. H76217060

LEMBAR PERSETUJUAN PEMBIMBING SKRIPSI

Skripsi Oleh

Nama : **Jeffri Dian Asmoro**

NIM : **H76217060**

Judul : ***VIRTUAL MOUSE* MENGGUNAKAN *HAND GESTURE RECOGNITION* BERBASIS *HAND LANDMARK MODEL* UNTUK *POINTING DEVICE***

Ini telah diperiksa dan disetujui untuk diujikan.

Surabaya, 12 Januari 2022

Menyetujui,

Dosen Pembimbing I	Dosen Pembimbing II
(Achmad Teguh Wibowo, MT)	(Mujib Ridwan, S.Kom., M.T)
NIP. 198810262014031003	NIP. 198604272014031004

LEMBAR PENGESAHAN TIM PENGUJI SKRIPSI

Skripsi Jeffri Dian Asmoro ini telah dipertahankan
di depan tim penguji skripsi
Surabaya, 14 Januari 2022

Mengesahkan,
Dewan Penguji

Penguji I

Dr. Eng. Anang Kunaefi, M.Kom
NIP. 197911132014031001

Penguji II

Bayu Adhi Nugroho, M.Kom
NIP. 197905182014031001

Penguji III

Achmad Teguh Wibowo, M.T
NIP. 198810262014031003

Penguji IV

Mujib Ridwan, S.Kom., M.T
NIP. 198604272014031004

Mengetahui,
Dekan Fakultas Sains dan Teknologi
UIN Sunan Ampel Surabaya

Prof. Dr. Hj. Evi Shofiatur Rusydiyah, M.Ag.
NIP. 197312272005012003

KEMENTERIAN AGAMA
UNIVERSITAS ISLAM NEGERI SUNAN AMPEL SURABAYA
PERPUSTAKAAN
Jl. Jend. A. Yani 117 Surabaya 60237 Telp. 031-8431972 Fax.031-8413300
E-Mail: perpus@uinsby.ac.id

LEMBAR PERNYATAAN PERSETUJUAN PUBLIKASI
KARYA ILMIAH UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademika UIN Sunan Ampel Surabaya, yang bertanda tangan di bawah ini, saya:

Nama : Jeffri Dian Asmoro
NIM : H76217060
Fakultas/Jurusan : Sains dan teknologi/Sistem Informasi
E-mail address : jeffridianasmoro@gmail.com

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Perpustakaan UIN Sunan Ampel Surabaya, Hak Bebas Royalti Non-Eksklusif atas karya ilmiah :
☐ Sekripsi ☐ Tesis ☐ Desertasi ☐ Lain-lain (.....)
yang berjudul :
VIRTUAL MOUSE MENGGUNAKAN *HAND GESTURE RECOGNITION* BERBASIS *HAND LANDMARK MODEL* UNTUK *POINTING DEVICE*

beserta perangkat yang diperlukan (bila ada). Dengan Hak Bebas Royalti Non-Ekslusif ini Perpustakaan UIN Sunan Ampel Surabaya berhak menyimpan, mengalih-media/format-kan, mengelolanya dalam bentuk pangkalan data (database), mendistribusikannya, dan menampilkan/mempublikasikannya di Internet atau media lain secara ***fulltext*** untuk kepentingan akademis tanpa perlu meminta ijin dari saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan atau penerbit yang bersangkutan.

Saya bersedia untuk menanggung secara pribadi, tanpa melibatkan pihak Perpustakaan UIN Sunan Ampel Surabaya, segala bentuk tuntutan hukum yang timbul atas pelanggaran Hak Cipta dalam karya ilmiah saya ini.

Demikian pernyataan ini yang saya buat dengan sebenarnya.

Surabaya, 23 Januari 2022
Penulis

(Jeffri Dian Asmoro)

ABSTRAK

VIRTUAL MOUSE MENGGUNAKAN *HAND GESTURE RECOGNITION* BERBASIS *HAND LANDMARK MODEL* UNTUK *POINTING DEVICE*

Oleh:

Jeffri Dian Asmoro

Saat ini teknologi semakin pesat dan sudah menjadi salah satu kebutuhan manusia yang harus dimiliki untuk menyelesaikan permasalahan yang sedang dihadapi. Perkembangan piranti masukan tanpa sentuhan atau *hand gesture recognition* menggunakan kamera adalah salah satu bentuk dari *machine learning*. Gestur mampu mendefinisikan sebagai gerakan fisik dari tangan, lengan, maupun badan sebagai pesan yang ekspresif, selain itu sistem gestur tangan ini mampu menjelaskan isi perintah yang memiliki arti. Dalam penelitian ini akan dikembangkan sebuah sistem *virtual mouse* menggunakan *hand gesture recognition* berbasis *hand landmark model* untuk *pointing device*. Aplikasi yang dihasilkan dapat dijalankan pada perangkat desktop dengan menggunakan webcam. Hasil dari pengujian yang dilakukan untuk menganalisa penerapan hand landmark model kedalam sistem menunjukkan rata-rata akurasi sistem mencapai 96% dan kecepatan mencapai 0.05 second.

Kata Kunci: ***Hand Gesture Recognition, Hand Landmark Model, Machine Learning, Virtual Mouse***

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

ABSTRACT

VIRTUAL MOUSE USING HAND GESTURE RECOGNITION BASED ON HAND LANDMARK MODEL FOR POINTING DEVICE

By:

Jeffri Dian Asmoro

Currently technology is growing rapidly and has become one of the human needs that must be possessed to solve the problems at hand. The development of touchless input devices or hand gesture recognition using a camera is a form of machine learning. Gestures are able to define as physical movements of the hands, arms, or body as expressive messages, besides that this hand gesture system is able to explain the contents of commands that have meaning. In this study, a virtual mouse system will be developed using hand gesture recognition based on a hand landmark model for pointing devices. The resulting application can be run on a desktop device using a webcam. The results of the tests carried out to analyze the application of the hand landmark model into the system show an average system accuracy of 96% and a speed of 0.05 seconds.

Kata Kunci: ***Hand Gesture Recognition, Hand Landmark Model, Machine Learning, Virtual Mouse***

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

DAFTAR ISI

LEMBAR PERNYATAAN KEASLIAN I
LEMBAR PERSETUJUAN PEMBIMBING SKRIPSI II
LEMBAR PENGESAHAN TIM PENGUJI SKRIPSI III
..... IV
KATA PENGANTAR V
ABSTRAK VI
DAFTAR ISI VIII
DAFTAR TABEL X
DAFTAR GAMBAR XI
BAB I 1
1. 1 Latar Belakang 1
1. 2 Rumusan Masalah 4
1. 3 Batasan Masalah 4
1. 4 Tujuan Penelitian 5
1. 5 Manfaat Penelitian 5
1. 6 Sistematika Penulisan 5
BAB II 7
2. 1 Penelitian Terdahulu 7
2. 2 Dasar Teori 9
2.2.1 Komputer Vision 9
2.2.2 Machine Learning 10
2.2.3 *Hand Gesture Recognition* 12
2.2.4 Single Shot Detector (SSD) 14
2.2.5 Pipeline Machine Learning 21
2.2.6 Palm Detector 22
2.2.7 Hand Landmark Model Menggunakan *Mediapipe* 22
2.2.8 *Open Source Computer Vision Library* 25
2.2.9 Python 26
2.2.10 Integrasi Keilmuan 27
BAB III 29
3.1 Perumusan Masalah 29
3.2 Studi Pustaka 30
3.3 Persiapan Alat dan Bahan 30
3.4 Pembuatan Sistem 30

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

3.5 Pengujian.....32
3.6 Evaluasi Hasil.....43
BAB IV.....44
4.1 Desain Arsitektur Sistem.....44
4.2 Alur Penggunaan Aplikasi.....45
4.3 UML Use Case Diagram.....46
4.4 UML Activity Diagram.....46
4.5 UML Squence Diagram.....49
4.6 Implementasi Sistem.....50
4.7 Pengujian Fungsionalitas Aplikasi.....54
4.8 Evaluasi Sistem.....62
BAB V.....64
5.1 Kesimpulan.....64
5.2 Saran dan Pemgembangan.....64
Daftar Pustaka.....66

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

DAFTAR TABEL

Tabel 3. 1 Skenario Pengujian Terhadap Jarak 33
Tabel 3. 2 Skenario Pengujian Terhadap Cahaya 36
Tabel 3. 3 Confusion Matrix 41
Tabel 3. 4 Test Case Scenario Pengujian Fungsionalitas Aplikasi 42
Tabel 4. 1 Hasil Pengujian Pengukuran Jarak 55
Tabel 4. 2 Hasil Pengujian Berdasarkan Pengukuran Cahaya 57
Tabel 4. 3 Hasil Pengujian Berdasarkan Test Case Scenario 61
Tabel 4. 4 Hasil Test Case Scenario 62
Tabel 4. 5 Tabel Hasil Evaluasi 62

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

DAFTAR GAMBAR

Gambar 2. 1 Flowchart Hand Gesture Recognition 13
Gambar 2. 2 (a) Kotak GT (b) Pola 8x8 (c) Pola 4x4 (Liu et al., 2016) 14
Gambar 2. 3 VGG-16 Network (Liu et al., 2016) 17
Gambar 2. 4 Hand Landmarks (Halder & Tayade, 2021) 23
Gambar 2. 5 Data Sintetis (Zhang et al., 2020) 25
Gambar 3. 1 Alur Penelitian 29
Gambar 3. 2 Alur Sistem Hand Gesture Recognition berbasis Hand Landmark Model 31
Gambar 4. 1 Desain Arsitektur Sistem 44
Gambar 4. 2 Alur Penggunaan Aplikasi 45
Gambar 4. 3 *Use Case Diagram* 46
Gambar 4. 4 *UML Activity Diagram Hand Landmark Model* 47
Gambar 4. 5 *UML Activity Diagram Gesture Recognition* 48
Gambar 4. 6 *UML Squence Diagram Hand Landmark Model* 49
Gambar 4. 7 *UML Squence Diagram Gesture Recognition* 50
Gambar 4. 8 Tampilan Aksi *Mouse Move* 53
Gambar 4. 9 Tampilan Aksi Klik Kanan 53
Gambar 4. 10 Tampilan Aksi Klik Kiri 54
Gambar 4. 11 Tampilan Aksi Double Klik 54

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

BAB I
PENDAHULUAN

1. 1 Latar Belakang

Teknologi terdiri dari *techne* yang berarti ‘keahlian’ dan *logia* yang berarti ‘pengetahuan’. Teknologi mengacu terhadap objek yang dipakai untuk memudahkan aktivitas manusia, contoh perkakas, mesin dan perangkat keras (Zulkarnaen, 2010). Pengertian lain, teknologi merupakan pengetahuan yang bertujuan menciptakan alat, Tindakan pengolahan dan untuk ekstraksi materi (Novitasari & Saleh, 2017).

Saat ini teknologi semakin pesat dan sudah menjadi salah satu kebutuhan manusia yang harus dimiliki untuk menyelesaikan permasalahan yang sedang dihadapi (Putra, 2017). teknologi berkembang pesat pada pemakaian komputer masa kini dengan ditujukan mampu meningkatkan efisiensi, karena dapat terbukti dengan menggunakan komputer maka pekerjaan dapat diselesaikan lebih efektif dalam segi biaya dan waktu (Indrajit, 2000).

Perkembangan teknologi yang semakin pesat tentunya juga diiringi dengan perubahan perilaku pada masyarakat yang lebih suka melakukan kegiatan dengan cepat dan praktis. Kemajuan teknologi merupakan fenomena yang tidak dapat terhindari. Setiap ide teknologi dihadirkan dengan tujuan memberikan dampak positif bagi kehidupan. Sampai saat ini manusia sudah menikmati banyak manfaat positif dari adanya teknologi (Ngafifi, 2014). Sehingga teknologi adalah penemuan benda maupun alat yang diwujudkan dari implementasi suatu ilmu pengetahuan serta keterampilan yang dipunyai oleh manusia.

Saat ini komputer masih sering digunakan dalam interaksi dengan manusia, sebagai contoh menekan tombol yang ada pada keyboard atau sentuhan dengan mouse. Sehingga masukan akan menghasilkan informasi sesuai dengan yang diinginkan. Tujuan utama dari interaksi manusia dan komputer adalah mempermudah pengguna saat mengoperasikan komputer. Prinsip kerja dari komputer atau teknologi adalah adanya masukan, proses dan menghasilkan output.

Pengoperasian komputer memerlukan piranti masukan. Piranti ini memiliki fungsi untuk memasukkan dan merekam data dalam system komputer dan

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

memberikan perintah. Piranti umum yang biasa dipakai untuk *personal computer* (PC) adalah keyboard atau mouse (Supal et al., 2020). Antarmuka mengalami perkembangan dari *command line* menjadi grafis dengan *keyboard* dan *mouse* sebagai media masukan. Selain itu *mouse* memiliki keunggulan yang lebih praktis serta cepat jika dibandingkan dengan *keyboard.* Mouse mulai dipakai secara maksimal setelah system operasi dengan basis *Graphical User Interface* (GUI). *Mouse* bekerja dengan bola kecil yang terdapat didalam mouse sehingga akan bergeser dan bergerak sesuai pergerakan yang dilakukan. Sensorik ball pada mouse akan berubah menjadi sinyal listrik sehingga mampu diproses oleh komputer. Tetapi mouse memiliki kecenderungan sering mengalami tracking yaitu meloncat-loncat, terkadang pula mouse tidak bisa di deteksi oleh PC, sehingga mouse tidak efisien, dari pembahasan tersebut pengguna membutuhkan antarmuka dengan komputer yang lebih mudah digunakan dan efisien (Murthy;Jadon, 2009). Salah satu solusi adalah memanfaatkan kamera sebagai piranti masukan secara *realtime* (Supal et al., 2020). Sehingga jika interaksi bisa dilakukan dengan efisien maka proses masukan akan bersifat efektif. Pemilihan piranti perlu memperhatikan factor psikologi kemudahan serta kebiasaan sehari-hari dengan saling berinteraksi alami saat memakai teknologi (Falabiba, 2019).

Saat ini pendekatan interaksi antara manusia dan komputer hanya memakai keyboard, mouse, dan pena. Sedangkan hal itu terasa tidak cukup dikarenakan perangkat ini memiliki keterbatasan. Penggunaan camera secara langsung sebagai perangkat masukan adalah metode yang menarik untuk menyediakan Interaksi manusia dan computer secara alami yang telah berkembang dari antarmuka berbasis teks melalui 2D antarmuka berbasis grafis, yang didukung multimedia, hingga sepenuhnya sistem Virtual Environment (VE) multi-peserta yang matang.

Perkembangan piranti masukan tanpa sentuhan atau *hand gesture recognition* menggunakan kamera sebagai pendeteksi gerakan tangan untuk mengenali gerakan tangan, sehingga interpretasi visual dari gestur tangan mampu memberikan efek kemudahan (Sunyoto & Harjoko, 2014). Gestur mampu mendefinisikan sebagai gerakan fisik dari tangan, lengan, maupun badan sebagai pesan yang ekspresif, selain itu system gestur tangan ini mampu menjelaskan isi perintah yang memiliki arti (Anwar et al., 2019).

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Selanjutnya, untuk penelitian kedepannya dalam *computer vision* penting untuk menetapkan system pengenalan gestur bertujuan untuk Interaksi Manusia dan Komputer (IMK) (Starner & Pentland, 1995). Tujuan utama lainnya sebagai system yang diciptakan khusus mengidentifikasi secara detail gestur manusia dan menggunakannya untuk perangkat control maupun penyampai informasi. Jadi dapat dikatakan *hand gesture recognition* merupakan teknologi yang beradaptasi dengan gestur tangan untuk sarana maupun penyampai informasi, dengan interpretasi gerakan memerintah semantic yang memiliki makna.

Pada penelitian (Garg et al., 2009) yang berjudul "*Vision Based Hand Gesture Recognition*" menjelaskan bahwa pemakaian *hand gestur recognition* dapat menggunakan metode Data-Glove ataupun computer vision. Perbedaanya terletak pada metode Data-Glove, yang membutuhkan perangkat sensor untuk mendigitalkan gerakan tangan menjadi data multiparametric, sehingga biayanya cukup mahal dan banyak mengalami kendala (Mulder, 1996).

Solusi untuk mengatasi permasalahan tersebut, maka dibutuhkan kamera untuk *Computer Vision* agar dapat menciptakan interaksi alami antara manusia dan computer tanpa perangkat tambahan (Velaskar et al., 2017). *Computer Vision* merupakan system dengan basis pengembangan algoritma dan computer untuk mensimulasikan visualisasi manusia agar secara otomatis dapat mengekstrak informasi dari harga suatu objek (Masithoh et al., 2012). *Computer vision* saat ini berkembang secara pesat seiring dengan meningkatnya kebutuhan manusia terhadap teknologi.

Saat ini, kecepatan pemrosesan data telah meningkat secara signifikan, sehingga dapat membantu manusia dalam tugas-tugas yang kompleks. Tetapi untuk teknologi masukan menyebabkan masalah baru dalam melakukan beberapa tugas misalkan lag, kurang memanfaatkan sumber daya yang tersedia dan membatasi ekspresi penggunaan aplikasi. Dengan begitu hand gesture recognition menjadi cara untuk mengatasinya. Pada penggunaan Hand Gesture Recognition menggunakan Hand Landmark Model. Hand Landmark Model berfungsi untuk mengalokasikan 21 titik pada tangan (Simon et al., 2017). Kelebihan dengan adanya Hand Landmark lebih mudah diimplementasikan sehingga memiliki output biner, dapat membedatakan tangan kanan dan kiri.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Langkah – langkah proses pada sistem ini menggunakan metode *Single Shot Detector* untuk mendeteksi dan memproses citra gambar menjadi numerik. Kemudian dalam mendeteksi tangan diperlukan *Hand landmark* sebagai prediksi yang memiliki 21 titik koordinat pada tangan, dari beberapa titik tersebut dipilih titik koordinat jari yang akan digunakan. Selanjutnya, proses dari diatas nantinya akan menggeluarkan *trigger*. Penelitian ini bertujuan untuk mengembangkan sistem *virtual mouse* secara *real-time* berbasis hand landmark model.

Berdasarkan penjelasan diatas teknologi *hand gesture recognition* dapat digunakan sebagai solusi teknologi untuk menggantikan mouse sebagai pointing device. Berawal dari latar belakang diatas, maka perlu dilakukan penelitian dengan judul "VIRTUAL MOUSE MENGGUNAKAN HAND GESTURE RECOGNITION BERBASIS HAND LANDMARK MODEL UNTUK POINTING DEVICE".

1. 2 Rumusan Masalah

Adapun perumusan masalah pada penelitian ini berdasarkan latar belakang diatas antara lain:

1. Bagaimana membuat teknologi *hand gesture recognition* menggunakan kamera?
2. Bagaimana penerapan *Hand Landmark Model* untuk *Virtual Mouse* menggunakan *hand gesture recognition*?
3. Bagaimana hasil evaluasi dalam penerapan *Hand Landmark Model* untuk *Virtual Mouse* menggunakan *hand gesture recognition*?

1. 3 Batasan Masalah

Agar penelitian ini dapat terfokuskan dan tidak menimbulkan hal yang tidak diharapkan maka dibuatlah batasan masalah sebagai berikut ini :

1. Penelitian ini untuk mengenalkan teknologi *Hand Gesture Recognition* menggunakan kamera.
2. Aplikasi ini menggunakan pemrograman Python.
3. Pada teknologi *Hand Gesture Recognition* menggunakan metode berbasis *Hand Landmark Model*.
4. Penelitian ini menggunakan pengujian Black Box, pengujian akurasi, dan

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

lux meter.

1. 4 Tujuan Penelitian

Penelitian memiliki tujuan antara lain:

1. Membuat virtual mouse menggunakan teknologi *hand gesture recognition* menggunakan kamera untuk *pointing device*.
2. Menerapkan *Hand landmark Model* untuk *Virtual Mouse* menggunakan *hand gesture recognition* untuk *pointing device.*
3. Mendapatkan hasil evaluasi dalam penerapan *Hand Landmark Model* untuk *Virtual Mouse* menggunakan *hand gesture recognition* untuk *ponting device.*

1. 5 Manfaat Penelitian

Manfaat dari adanya riset ini memiliki dampak yang luas terhadap pihak maupun lingkungan yang terkait pada bidang ini, termasuk terhadap akademisi ataupun secara aplikatif. Adapun manfatnya didapatkan sebagaimana berikut :

1. Akademis

   Mampu untuk mengimplemantasi ilmu yang sudah didapatkan guna menyelesaikan masalah yang terjadi pada masyarakat. Selain itu, dapat memberikan pengetahuan baru dengan mengembangkan metode yang sudah ada sebelumnya terhadap inovasi pada aplikasi.
2. Aplikatif

   Dengan adanya *software* ini memudahkan masyarakat umum terutama sebagai pengguna komputer pada saat menggunakan mouse digantikan dengan tangan dan berinteraksi secara alami dan real time yang terkait dan juga bisa mendapat ilmu secara cepat hingga tepat

1. 6 Sistematika Penulisan

Guna memahami penelitian ini , dilakukan dengan cara mengelompokkan materi menjadi beberapa sub bab dengan sistematika penulisan sebagai berikut :

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

- BAB I PENDAHULUAN

  Bab 1 terdiri dari latar belakang permasalahan, rumusan masalah yang didapatkan dari latar belakang sehingga menghasilkan tujuan dari riset ini, manfaat penelitian hingga batasan dan sistematika kepenulisan
- BAB 2 TINJAUAN PUSTAKA

  Bab 2 terdiri dari penelitian terdahulu yang releven, dasar teori hingga integrase kelimuan dari penelitian ini.
- BAB 3 METODOLOGI PENELITIAN

  Bab 3 terdiri dari model riset dan metode yang dipakai pada penelitian ini.
- BAB 4 HASIL DAN PEMBAHASAN

  Bab 4 terdiri dari hasil penyusunan program yang ada pada riset ini.
- BAB 5 PENUTUP

  Bab 5 terdiri dari kesimpulan dan saran dari penelitian ini setelah melaksanakan beberapa tahap riset.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

BAB II
STUDI PUSTAKA

2. 1 Penelitian Terdahulu

Penelitian terdahulu menjadi dasar pedoman untuk menjalankan penelitian, sehingga memperdalam teori yang dipakai untuk mengkaji riset ini. Pada penelitian ini terdapat beberapa penelitian terdahulu dengan menggunakan berbagai metode dan penerapan yang berbeda, salah satunya yang berjudul "Hand Gesture Recognition Sebagai Pengganti Mouse Komputer Menggunakan Kamera". Metode yang digunakan dengan *convexhull algorithm*, yaitu sebuah metode dengan menjadikan jumlah jari tangan yang nantinya akan menjadi acuan untuk pengerjaan aksi mouse. Dalam penerapan metode ini masih memiliki kekurangan yaitu metode convexhull algorithm masih memiliki kendala untuk membedakan objek yang sama dengan warna kulit dan juga background yang tidak teratur (Yunita & Setyati, 2019).

Selanjutnya pada penelitian (Halder & Tayade, 2021) dengan judul "Real-time Vernacular Sign Language Recognition using MediaPipe and Machine Learning" dengan menggunakan *Hand Landmark Model* dan *Palm Detector*. *Palm Detector* mendeteksi tangan secara akurat yang nantinya akan diteruskan ke *Hand Landmark Detector*. Dengan mendapatkan hasil persentase akurasi sebesar 99%. Kesimpulan dari penelitian ini adalah deteksi akurat *real-time* menggunakan *Hand Landmark Model* dan *Palm Detector* tanpa sensor dapat diterapkan dalam menggunakan teknologi ini lebih efektif dan efisien. Kelebihan lainnya dapat mengurangi proses augmentasi data yaitu rotasi, flipping dan scaling.

Riset selanjutnya oleh (Yuliana et al., 2016) dengan judul "Metode Convex Hull dan Convexity Defects untuk Pengenalan Isyarat Tangan". Metode yang digunakan dengan convex Hull dan convexity Defects. Metode convex hull untuk mendeteksi titik Hull sebagai ujung jari dan convexity defects mencari deskripi jari. Pada penerapannya masih memliki keterbatasan hanya menggunakan tangan kiri dan background tidak teratur ataupun background yang memiliki warna kulit masih sulit untuk dapat terdeteksi. Hal ini pun disebabkan

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

karena proses pencahayaan yang berlebihan sehingga deteksi menjadi tidak stabil.

Dalam penerapan support vector machine pada penelitian (Neneng et al., 2016) dengan judul "Support vector machine untuk klasifikasi citra jenis daging berdasarkan tekstur menggunakan ekstraksi ciri Gray Level C0 Occurance Matrices (GLCM)" dengan menggunakan metode GLCM menunjukan bukti bahwasannya tekstur menjadi descriptor paling kuat untuk mendeteksi sebuah citra. Dengan hasil jarak terbaik untuk sebuah pengenalan citra dengan jarak 20 cm dengan tingkat pengenalan terbaik yaitu 87,5%.

Berdasarkan (Islam et al., 2017) yang berjudul "Real Time Hand Gesture recognition Using Different Algorithms Based on American Sign Language" metode yang digunakan dengan metode gabungan yaitu K Convex Hul dengan metode gabungan ini diperoleh tingkat akurasi pengenalan sangat tinggi dan tajam yaitu sebesar 94,23%. Akan tetapi untuk metode ini memiliki kekurangan hanya bisa dideteksi dengan menggunakan background hitam sehingga dengan begitu dapat meminimalisir kekeliruan saat pengenalan gestur.

Kemudian pada penelitian (Rokade et al., 2016) dengan judul "Hand Gesture Recognition System For Multimedia Applications" dengan menggunaakan metode kinect. Metode Kinect merupakan metode interaksi manusia dan computer agar dapat berjalan secara natural. Dengan hasil penelitian yaitu melalui mengkontrol multiple aplikasi seperti pembaca pdf, dengan menggunakan kamera web. Akan tetapi penelitian ini memiliki keterbatasan yaitu kamera hanya dapat membaca pdf sehingga manusia tidak dapat berinteraksi langsung dengan touchscreen.

Setelah itu pada penelitian (K V et al., 2020) dengan judul "Hand Gesture Recogition and Voice Conversion for Hearling and Speech Aided Community" dengan menggunakan metode *Support Vector Machine, K neighbors Classifier, Logistic Regression, MLP Classifier, naïve Bayes, Random Forest Classifoer algoritm*. Dengan menggunakan metode itu dapat merancang sebuah sistem untuk membantu orang yang melatih tunarungu untuk berkomunikasi dengan seluruh dunia menggunakan Bahasa isyarat atau teknik pengenalan gerakan

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

tangan.

Selanjutnya pada penelitian (Munawarah et al., 2016) dengan judul "Penerapan Metode Support Vector Machine Pada Diagnosa Hepatitis" dengan menerapkan metode *Support Vector Machine*. Hasil kemampuan analisis metode diketahui dengan pengujian menggunakan data testing melalui kedua kernel dengan data training 100 data positif dan 100 data negatif. Hasil pengujian yang menggunakan fungsi kernel linier mendapatkan hasil persentase sebesar 68-83 % dan fungsi kernel RBF 70-96 %.

Setelah itu pada penelitian terdahulu yang terakhir adalah "SSd: Single Shot Multibox detector" dengan penulis (Liu et al., 2016) dengan metode SSD, Dataset PASCAL, VOC, COCO dan ILSVRC. Dengan metode SSD sederhana dan membutuhkan objek selanjutnya terdapat tahapan resampling piksel dan fitur merangkum dalam satu jaringan sehingga dengan metode tersebut dapat mengurangi terjadinya kesalahan pendeteksian. Dibandingkan dengan metode satu tahap lainnya, SSD memiliki akurasi yang jauh lebih baik bahkan dengan ukuran gambar input yang lebih kecil.

Dari beberapa referensi penelitian terdahulu dapat disimpulkan untuk penelitian ini menggunakan SSD karena lebih mudah diterapkan dan menggunakan *Hand Landmark Model* dan *Palm Detector* agar proses system lebih cepat teratasi dan dipakai selain itu tidak membutuhkan training sehingga lebih efisien.

2. 2 **Dasar Teori**

2.2.1 Komputer Vision

Computer Vision System (CVS) memiliki tujuan untuk computer agar dapat memahami video dan citra (Szeliski, 2010). Pada penelitian yang sudah dilakukan yaitu: penambahangan data berupa text (2), Perbaikan citra digital dan magnifikasi dan amplifikasi gerakan mampu bekerja pada saat memroses gambar digital dengan khusus, serta mampu menangkap gerakan wajah tanpa ataupun sedikit penanda adalah salah satu tehnik baru yang ada pada *computer vision* (Defri, 2016)

CVS merupakan system dengan basis pengembangan algoritma dan juga

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

computer untuk menstimulasikan visualisasi manusia supaya bisa secara otomatis mengekstrak informasi dari harga suatu objek (Masithoh et al., 2012). Dibandingkan metode konvensional yang membutuhkan waktu lama dengan membutuhkan analisa laboratorium yang rumit, maka CVS mampu menjadi suatu alternatif metode non-destruktif untuk mengamati dan mengukur kualitas. Computer visi merupakan cabang dari sebuah Teknik kecerdasan buatan (*artificial intelligence*) yang memiliki hubungan dengan kegiatan simulasi visual manusia.

CVS memiliki komponen dari otak manusia, sensor, maupun kamera digital dan juga system pererangan berupa sumber dari cahaya untuk difasilitasi akuisisi citra (*image acquisition*), pengolahan citra (*image processing*), dan penerjemahan citra (*image understanding*). Secara lebih rinci, tahapan dalam analisa citra adalah: 1) pembentukan citra, dimana citra suatu objek diambil dan disimpan dalam computer, 2) pre proses citra, dimana kualitas citra digital mampu diperbaiki untuk meningkatkan detil citra, 3) segmentasi citra, dimana citra objek diidentifikasi dan dipisahkan dari latar belakangnya, 4) pengukuran citra, dimana beberapa fitur-fitur yang sudah diekstrak kemudian dapat diinterpretasikan.

2.2.2 Machine Learning

Pembelajaran mesin dengan pendekatan artificial intelligence dapat digunakan untuk mengganti dan meniru perilaku manusia untuk pemecahan masalah dialkukan secara otomatisasi sering disebut dengan Machine Learning atau ML. ML mencoba untuk meniru alur proses makhluk cerdas atau manusia dengan generalisasi dan belajar. Singkatnya machine learning dapat membangun perhitungan dengan kemungkinan suatu program computer melakukan job secara otomatis dan mandiri. Algoritma sejenis ini mampu membangun suatu model dari masukan untuk bisa menghasilkan sebuah prediksi maupun pengambilan keputusan dengan dasar data yang ada. Penerapan ML dapat digunakan image processing dan text analysis. Terdapat 3 kategori utama pada machine learning, sebagai berikut:

a. *Supervised Learning*, yaitu data yang memiliki perlengkapan suatu label/kelas untuk menunjukan klasifikasi maupun kelompok dengan data berbeda. Model yang dihasilkan dapat berupa model prediksi dengan data

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

yang sudah diberi label. Algoritma yang digunakan untuk Teknik supervised learning diantarnya decision tree, naïve bayes, regresi, K-Nearest Neighboor (KNN) dan Super Vector Machine (Sodik et al., 2020).

b. *Unsupervised Learning,* yaitu tidak adanya label/kelas pada data pembelajaran yang pada akhirnya diharuskan untuk mencari data yang memiliki alur struktur didalamnya, kemudian dari informasi yang sudah ada akan melewati proses untuk dikelompokkan. Algoritma yang digunakan untuk Teknik unsupervised learning diantaranya Fuzzy C-Means, border, noise, K-Means, DBSCAN, self organizing map (Ashari et al., 2019).

c. *Reinforcement Learning,* yaitu pembelajaran dengan memetakan situasi terhadap aksi untuk memperoleh reward secara maksimal. Pembelajar akan menampilkan aksi yang bisa memberikan hadiah secara maksimal dengan cara menjalankannya.

Selain itu tugas dari ML ialah untuk klasifikasi semisal ingin menentukan sebuah hal dengan system, maka hal yang dapat dilakukan dengan menimbang atau dapat memakai kompter vision untuk mengenali suatu bentuk dan hal lain guna mengumpulkan informasi. Jika informasi sudah terkumpul, maka hal terakhir yang dapat dilakukan dengan klasifikasi yang nanti akan menghasilkan suatu output berupa tipe klasifikasi yang diinginkan. Tentunya tahap klasifikasi dengan menggunakan algoritma diatas seperti supervised learning ataupun unsupervised learning. Setelah bisa menggunakan algoritma pada ML untuk klasifikasi maka Langkah selanjutnya dengan melatih algoritma tersebut, untuk melatih algoritma dapat digunakan dengan cara training atau dapat dikenal dengan training set. Pada setiap training memiliki variable dan fitur target. Target pada variable ialah apa yang akan diprediksikan oleh algoritma ML, yang nantinya ML akan mmpelajari hubungan antar variable target dan fitur. ML memiliki tugas untuk klasifikasi tetapi dalam prosesnya tejadi penentuan class untuk sebuah instance berada yaitu untuk regresi. Regresi ialah memprediksi dari nilai numerik. Regresi termasuk dalam tipe algoritma dengan pembelajaran terarah.

Untuk pengembangan aplikasi pada ML, dapat dijelaskan sebagai berikut:

1. Mengumpulkan Data, proses pengumpulan data dilakukan dengan cara memberi contoh sampel dari berbagai informan, yang ada di internet dan

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

media lainnya. Data yang terkumpul ialah data yang sudah disebar secara bebas ke umum.

2. Mempersiapkan Data Masukan, pada proses ini data masukan yang sudah disiapkan ialah data masukan sesuai dengan kualifikasi dari kebutuhan yang sudah dianalisa.
3. Melatih Algoritma, pada tahap ini pemakai dengan data yang memiliki kualitas dan nanti algoritma akan mengolah sehingga akan diubah menjadi informasi sekaligus disimpan.
4. Menguji Algoritma, pada tahap ini dengan melihat seberapa baik dari kualitas algoritma yang sudah dilatih sebelumnya.
5. Menggunakannya, pada tahap ini dengan adanya algoritma yang sudah diterapkan pada suatu program, sehingga bisa melakukan suatu perintah yang diinginkan, dan setelah itu dilakukan pengecekan ulang terhadap Langkah sebelumnya.

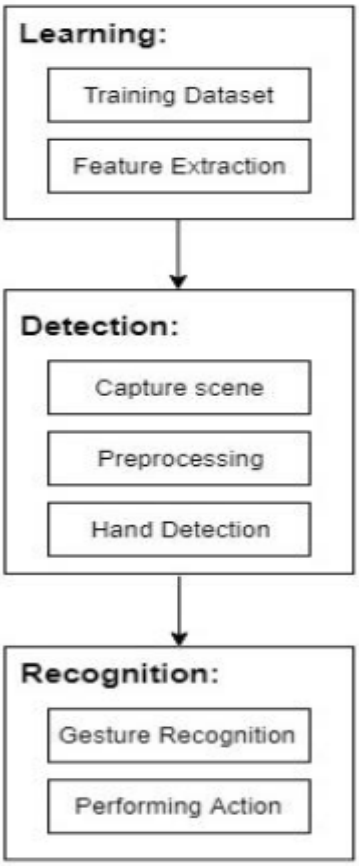
2.2.3 *Hand Gesture Recognition*

Peningkatan daya komputasi telah membawa banyak perubahan perangkat dalam kehidupan sehari-hari manusia. Sebuah spektrum yang luas dari aplikasi dan antarmuka telah dikembangkan sehingga manusia dapat berinteraksi dengan komputer. Interaksi dengan sistem ini lebih mudah karena cenderung dilakukan dengan cara alami (yaitu, seperti manusia berinteraksi satu sama lain menggunakan suara atau gerakan). *Hand Gesture Recognition* (HGR) adalah elemen penting dari *Human Computer Interaction* (HCI), yang mempelajari teknologi komputer yang dirancang untuk menginterpretasikan perintah yang diberikan oleh manusia. Model HGR telah banyak digunakan, misalnya, dalam beberapa aplikasi, seperti prostesis cerdas (Shi et al., 2018), pengenalan bahasa isyarat (Islam et al., 2017), rehabilitasi perangkat (Nelson et al., 2018), dan kontrol perangkat (Sarkar et al., 2016).

Dengan arti lain HGR adalah teknik yang digunakan untuk memahami dan menganalisis bahasa tubuh manusia dan berinteraksi dengan pengguna yang sesuai. Hal ini mampu membantu dalam membangun jembatan antara mesin dan pengguna untuk saling berkomunikasi. Pengenalan gerakan berguna dalam memproses informasi yang tidak dapat disampaikan melalui ucapan atau teks.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Gestur adalah cara paling sederhana untuk mengkomunikasikan antara manusia dan komputer. Implementasi HGR yang bertujuan untuk merancang gerakan tangan berbasis visi sistem pengenalan dengan tingkat deteksi yang tinggi bersama dengan kriteria kinerja tinggi, yang dapat bekerja secara real time yaitu Interaksi Manusia Komputer tanpa batasan (sarung tangan, seragam latar belakang dll.) pada lingkungan pengguna. Sistem dapat didefinisikan menggunakan flowchart seperti yang ditunjukkan pada Gambar 1.1.



Gambar 2. 1 Flowchart Hand Gesture Recognition

Pada Gambar 2.1 dapat dijelaskan bahwa Langkah dari sistem HGR dengan 3 tahap utama yaitu, *Learning, Detection* dan *Recognition* untuk lebih jelasnya sebagai berikut:

1. Learning dengan melibatkan 2 proses yaitu dataset pelatihan dan ekstraksi fitur. Dataset pelatihan adalah kumpulan data yang terdiri dari berbagai jenis gerakan tangan yang digunakan untuk melatih sistem berdasarkan tempat sistem melakukan tindakan. Ekstraksi Fitur melibatkan penentuan centroid yang membagi gambar menjadi dua bagian di Pusat geometrisnya.

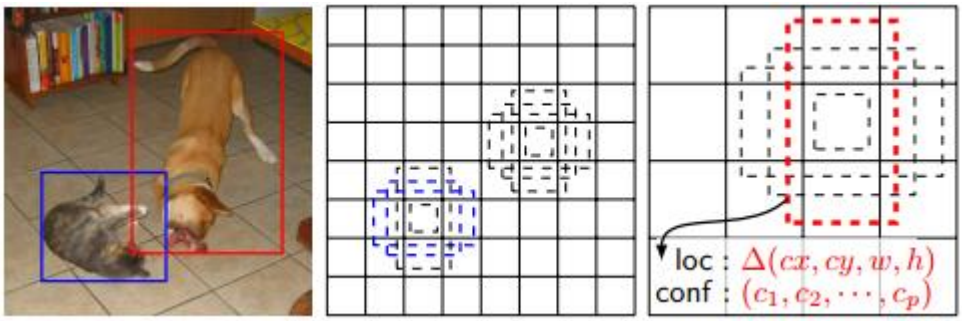
digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

2. Deteksi dengan melibatkan 3 proses yaitu tangkap scene, prapemrosesan dan deteksi tangan atau gestur. Tangkap adegan adalah menangkap gambar melalui kamera web, yang digunakan sebagai masukan ke sistem. Prapemrosesan ialah gambar yang diambil melalui webcam dibandingkan dengan dataset untuk mengenali gerakan tangan yang valid yang diperlukan untuk melakukan tindakan yang diperlukan. Deteksi Tangan ialah persyaratan untuk deteksi tangan melibatkan gambar masukan dari webcam. Gambar harus diambil dengan kecepatan 20 frame per detik. Jarak juga harus dijaga antara tangan dan kamera. Perkiraan jarak yang seharusnya antara tangan kamera adalah sekitar 30 sampai 100 cm. Masukan video disimpan frame demi frame ke dalam matriks setelah preprocessing.
3. Recognition melibatkan Pengenalan Gerakan dan melakukan Tindakan. Pengenalan gerakan yaitu Jumlah jari yang ada dalam gerakan tangan ditentukan dengan memanfaatkan titik cacat hadir dalam gerakan. Gerakan yang dihasilkan yang diperoleh dimasukkan melalui *Hand Landmark Model* secara berurutan untuk mengenali gerakan saat ini. Melakukan Tindakan yaitu gerakan yang dikenali digunakan sebagai masukan untuk melakukan tindakan yang diperlukan oleh pengguna.

2.2.4 Single Shot Detector (SSD)

Pada bagian ini akan dijelaskan mengenai kerangka kerja atau *framework* dari SSD, model, training methodology. Untuk lebih detailnya akan dipaparkan sebagai berikut:

a. Framework SSD



a.Gambar dengan Kotak GT b. Pola 8 x 8 c. Pola 4 x 4

Gambar 2. 2 (a) Kotak GT (b) Pola 8x8 (c) Pola 4x4 (Liu et al., 2016)

Pada Gambar 2.2 dijelasakan bahwa terdapat 3 gambar a,b dan c. Pada

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Gambar a dapat diketahui bahwa SSD hanya membutuhkan gambar masukan dan kotak dasar untuk setiap objek dalam masa training sehingga lebih mudah dipakai. Dalam mode konvolusi, dapat dievaluasi menjadi set kecil dengan kotak default dengan rasio aspek yang berbeda pada setiap lokasi di beberapa pola dengan skala yang berbeda missal 8 x 8 dan 4 x 4 pada gambar b dan c. untuk setiap kotak default, system mampu memprediksi baik offset bentuk dan kepercayaan untuk semua kategori objek ((c1;c2;…;cp)). Pada waktu training, pertama yang dilakukan adalah mencocokkan kotak dengan standar ini dengan dasar kotak kebenaran. Sebagai contoh, pada Gambar 1 terdapat default dua gambar dengan mencocokan gambar satu kucing dan anjing, yang diperlakukan sebagai positif dan sisanya sebagai negative (Girshick, 2015).

Salah satu cara mendeteksi dengan bidikan tunggal yaitu dapat menggunakan SSD. SSD dapat menjadi pilihan utama dikarenakan SSD lebih unggul daripada detector bidikan tunggal lainnya yaitu YOLO. SSD lebih cepat, akurat dan signifikan (Liu et al., 2016). Tingkat akurasi dari SSD ini sama dengan eksplisit dan penyatuan (Faster R-CNN). Inti dari SSD adalah untuk memprediksi skor kategori dan kotak pembatas akan tetapi untuk kotak pembatas masih default menggunakan filter konvolusi kecil yang diterapkan pada peta fitur. Untuk mencapai akurasi deteksi yang tinggi, maka perlu menghasilkan prediksi skala yang berbeda dari menampilkan peta, secara eksplisit mampu memisahkan prediksi berdasarkan rasio aspek. Fitur desain ini mampu menghasilkan training end-to-end yang sederhana dengan akurasi tinggi, bahkan pada gambar masukan memiliki resolusi rendah, lebih mampu meningkatkan kecepatan versi akurasi trade-off.

b. Model

Pendekatan SSD didasarkan pada jaringan konvolusi feed forward yang mengahasilkan kumpulan kotak pembatas. Lapisan awal jaringan didasarkan pada arsitektur standar yang dipakai untuk klasifikasi gambar dengan kualitas tinggi, berfungsi untuk memanggil jaringan dasar. Terdapat struktur tambahan kedalam jaringan dengan menghasilkan deteksi dengan fitur sebagai berikut:

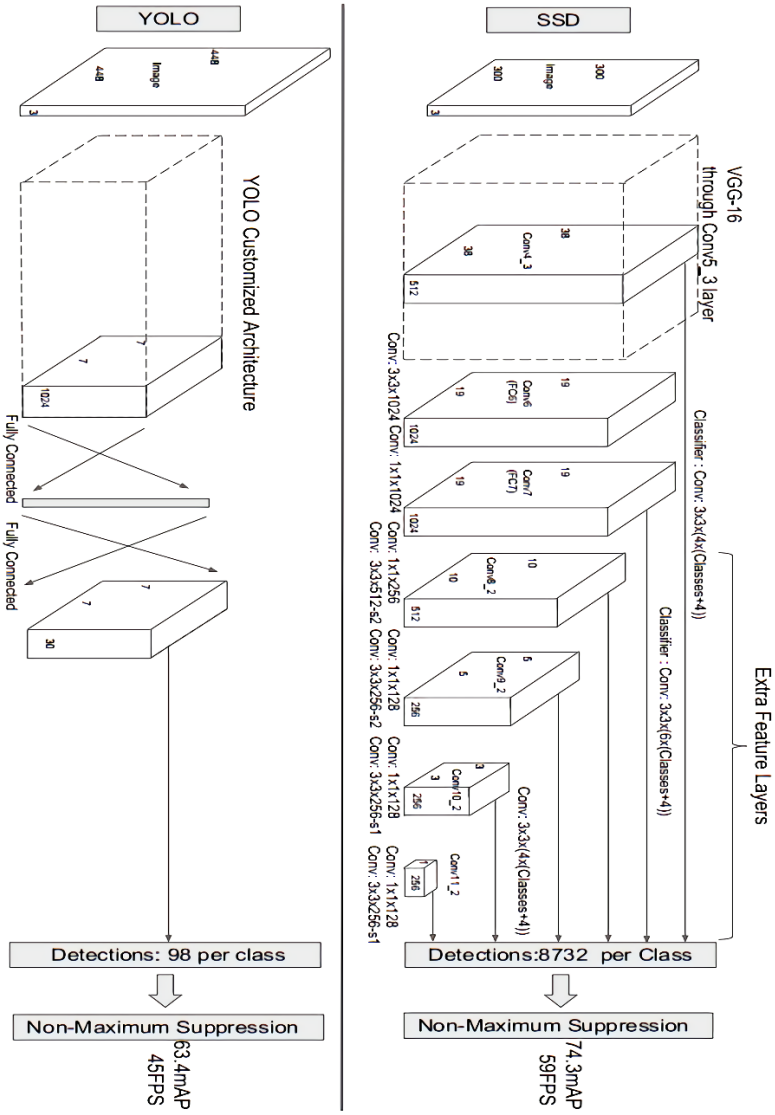
1. Peta fitur multi-skala, untuk mendeteksi perlu penambahan fitur konvolusi sampai akhir dari jaringan terpotong. Lapisan-lapisan ini

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

semakin mengecil ukurannya dan memungkinkan prediksi deteksi pada berbagai skala. Model konvolusi untuk memprediksi deteksi berbeda untuk setiap lapisan fitur (Sermanet et al., 2014).

2. Prediktor konvolusional untuk deteksi pada setiap lapisan fitur yang bisa ditambahkan secara opsional pada jaringan dasar mampu menghasilkan serangkaian prediksi deteksi dengan tetap menggunakan satu set filter konvolusi. Hal ini bisa ditunjukkan pada Gambar 2.3 dibawah yang merupakan jaringan SSD arsitektur. Pada lapisan fitur ukuran m x n dengan saluran p, elemen dasar untuk memprediksi parameter deteksi potensial adalah kernel kecil 3 x 3 x p yang menghasilkan skor digunakan suatu kategori, atau bentuk offset relative terhadap kotak default koordinat. Pada masing-masing lokasi m x n dimana kernel diterapkan, menghasilkan nilai keluaran.nilai output offset kotak pembatas diukur relative terhadap default.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id



Gambar 2. 3 VGG-16 Network (Liu et al., 2016)

Pada Gambar 2.3 Dapat dilihat perbandingan antara dua model deteksi tembakan tunggal: SSD dan YOLO (Redmon et al., 2016). Model SSD kami menambahkan beberapa lapisan fitur ke ujung jaringan dasar, yang memprediksi offset ke kotak default dari berbagai skala dan rasio aspek serta kepercayaan terkaitnya. SSD dengan ukuran input 300 × 300 secara signifikan mengungguli jenis tipe 448 × 448 YOLO dalam akurasi pada pengujian VOC2007 sekaligus meningkatkan kecepatan. Posisi kotak relative terhadap setiap lokasi peta fitur arsitektur YOLO (Redmon et al., 2016) yang menggunakan lapisan perantara terhubung penuh alih-alih filter konvolusi.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

3. Kotak default dan rasio aspek mampu mengaitkan satu set kotak pembatas default dengan setiap sel peta fitur, untuk beberapa peta fitur dibagian atas jaringan. Standartnya kotak peta fitur secara konvolusi, sehingga posisi setiap kotak relative terhadap sel yang sesuai. Di setiap sel peta fitur, kami memprediksi offset relative terhadap bentuk kotak defaut di sel, serta skor kelas yang mampu menunjukan instance pada masing-masing kotak. Khususnya untuk setiap kotak dari k di lokasi tertentu, dapat menghitung skor kelas c dan 4 offset relative terhadap aslinya dengan bentuk kotak standar. selanjutnya menghasilkan (c + 4) keluaran untuk peta fitur m x n. Untuk ilustrasi kotak default, bisa dilihat pada Gambar 1. Kotak default tersebut mirip dngan kotak jangkar yang digunakan di Faster R-CNN (Ren et al., 2017), namun mampu diterapkan ke beberapa fitur peta resolusi yang berbeda. Mengizinkan berbagai bentuk kotak default di beberpa fitur peta memmungkinkan kita secara efisien membedakan ruang dari kemungkinan bentuk kotak keluaran.

c. Training Methodology

Perbedaan utama antara melatih SSD dan melatih detector adalah bahwa informasi kebenaran dasar ditetapkan ke output tertentu di set tetap keluaran detector. Beberapa versi juga diperlukan pelatihan di YOLO (Redmon et al., 2016), tahap proposal region Faster R-CNN (Ren et al., 2017) dan multibox (Erhan et al., 2014). Setiap kali penugasan ditentukan, kerugian dan propagasi balik diterapkan dari ujung ke ujung. Training juga melibatkan pemilihan set kotak dan skala default untuk mendeteksi serta penambangan negative keras dan strategi sugmentasi data.

1. Strategi Pencocokan selama pelatihan, perlu menentukan kotak default mana yang sesuai dengan deteksi dasar dan melatih jaringan yang sesuai. Untuk setiap kotak yang dipilih dari kotak default dapat bervariasi berdasarkan lokasi, rasio aspek, dan skala. Mulai dengan mencocokkan setiap kotak dasar ke kotak default dengan yang terbaik menggunakan jaccard (seperti dalam MultiBox). Tidak seperti MultiBox, kemudian mencocokkan kotak default dengan setiap kebenaran dasar dengan jaccard tumpang tindih lebih tinggi dari ambang (0,5). Hal ini

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

menyederhanakan masalah pembelajaran, memungkinkan jaringan untuk memprediksi skor tinggi untuk beberapa tumpang tindih kotak default daripada mengharuskannya untuk memilih hanya satu dengan tumpang tindih maksimum.

2. Tujuan pelatihan SSD merupakan turunan dari tujuan MultiBox, tetapi dimodifikasi dengan memperluas untuk menangani beberapa kategori objek. Misal xp ij = f1; 0g jadilah indikator untuk mencocokkan kotak default ke-i dengan kotak kebenaran dasar ke-j dari kategori p. Dalam strategi pencocokan di atas, dapat memiliki Pi xp ij 1. Kerugian pasti terjadi jika keseluruhan fungsi adalah jumlah tertimbang dari kehilangan lokalisasi (loc) dan kehilangan kepercayaan (conf):

$$L(x,c,l,g) = \frac{1}{N}\left(L_{conf}(x,c) + \alpha L_{loc}(x,l,g)\right) \qquad (1)$$

di mana N adalah jumlah kotak default yang cocok. Jika N = 0, wet atur loss ke 0. Kerugian lokalisasi adalah kerugian L1 Halus antara kotak yang diprediksi (l) dan tanah parameter kotak kebenaran (g). Mirip dengan Faster R-CNN, kami mundur ke offset untuk pusat (cx; cy) dari kotak pembatas default (d) dan untuk lebarnya (w) dan tingginya (h).

$$L_{loc}(x,l,g) = \sum_{i \in Pos}^{N} \sum_{m \in \{cx,cy,w,h\}} x_{ij}^{k} smooth_{L1}(l_i^m - \hat{g}_j^m)$$

$$\hat{g}_j^{cx} = \left(g_j^{cx} - d_i^{cx}\right) / d_i^w \quad \hat{g}_j^{cy} = \left(g_j^{cy} - d_i^{cy}\right) / d_i^h$$

$$\hat{g}_j^w = \log\left(\frac{g_j^w}{d_i^w}\right) \qquad \hat{g}_j^h = \log\left(\frac{g_j^h}{d_i^h}\right) \qquad (2)$$

Kehilangan kepercayaan adalah kerugian softmax atas beberapa kepercayaan kelas (c)

$$L_{conf}(x,c) = -\sum_{i \in Pos}^{N} x_{ij}^{p} \log\left(\hat{c}_i^p\right) -$$

$$\sum_{i \in Neg} \log\left(\hat{c}_i^0\right) \; where \; \hat{c}_i^p = \frac{\exp\left(\hat{c}_i^p\right)}{\sum_p \exp\left(c_i^p\right)} \qquad (3)$$

dan istilah bobot disetel ke 1 dengan validasi silang.

3. Memilih skala dan rasio aspek untuk kotak default Untuk menangani skala objek yang berbeda-beberapa metode menyarankan pemrosesan

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

gambar pada ukuran yang berbeda dan menggabungkan hasil setelahnya. Namun, dengan memanfaatkan peta fitur dari beberapa lapisan yang berbeda dalam jaringan tunggal untuk prediksi dapat meniru efek yang sama, sementara juga berbagi parameter di semua skala objek. Karya-karya sebelumnya telah menunjukkan bahwa menggunakan peta fitur dari lapisan bawah dapat meningkatkan kualitas segmentasi semantik karena semakin rendah lapisan menangkap detail yang lebih halus dari objek input. Demikian pula, menunjukkan bahwa menambahkan konteks global yang dikumpulkan dari peta fitur dapat membantu memperlancar hasil segmentasi.

Metode ini, menggunakan peta fitur bawah dan atas untuk deteksi. Gambar 2.2 menunjukkan dua contoh peta fitur (8x8 dan 4x4) yang digunakan dalam kerangka. Dalam praktiknya, kita dapat menggunakan lebih banyak lagi dengan overhead komputasi yang kecil. Peta fitur dari tingkat yang berbeda dalam jaringan diketahui memiliki perbedaan (empiris) ukuran bidang reseptif. Untungnya, dalam kerangka kerja SSD, kotak default tidak perlu harus sesuai dengan bidang reseptif yang sebenarnya dari masing-masing kotak lapisan. Kami merancang ubin kotak default sehingga peta fitur tertentu belajar menjadi responsif terhadap skala tertentu dari objek. Misalkan kita ingin menggunakan peta fitur untuk prediksi.

4. Penambangan negatif keras Setelah langkah pencocokan, sebagian besar kotak default adalah negatif, terutama bila jumlah kotak default yang mungkin besar. Ini memperkenalkan ketidakseimbangan yang signifikan antara contoh pelatihan positif dan negatif. Dari pada menggunakan semua contoh negatif, mengurutkannya menggunakan kehilangan kepercayaan tertinggi untuk masing-masing kotak default dan pilih yang teratas sehingga rasio antara negatif dan positif adalah paling banyak 3:1. Dapat dijelaskan bahwa akan mengarah pada pengoptimalan yang lebih cepat dan pelatihan yang lebih stabil
5. Augmentasi data untuk membuat model lebih kuat terhadap berbagai ukuran objek input dan bentuk, setiap gambar pelatihan diambil

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

sampelnya secara acak oleh salah satu opsi berikut:

- Gunakan seluruh gambar masukan asli.
- Contohkan patch sehingga jaccard minimum yang tumpang tindih dengan objek adalah 0,1, 0,3,0,5, 0,7, atau 0,9.
- Sampel tambalan secara acak

Ukuran setiap patch sampel adalah [0,1, 1] dari ukuran gambar asli, dan rasio aspek berada di antara 1 dan 2. Kami menyimpan bagian kotak kebenaran dasar yang tumpang tindih jika pusatnya itu ada di patch sampel. Setelah langkah pengambilan sampel yang disebutkan di atas, setiap tambalan sampel diubah ukurannya menjadi ukuran tetap dan dibalik secara horizontal dengan probabilitas 0,5.

2.2.5 Pipeline Machine Learning

Pipeline Machine Learning (PML) merupakan alur kerja yang dapat dijalankan dengan independent dari beberapa pembelajaran mesin yang sudah lengkap. Serangkaian dalam PML dibagi menjadi beberapa subtugas yang di enkapsulasi (Derakhshan et al., 2019). PML dapat menggunakan bahasa Python. Terdapat beberapa fokus tugas dari adanya PML seperti:

1. Mempersiapkan data termasuk didalamnya import, validasi, munging, transformasi, normalisasi dan staging
2. Konfigurasi pelatihan termasuk kedalam argument parameterisasi, jalur file dan konfigurasi logging/pelaporan.
3. Melatih dan memvalidasi secara efisien dan berulang kali. Efisiensi berasal dari menentukan subset data tertentu, sumber daya komputasi perangkat keras yang berbeda, pemrosesan terdistribusi dan pemantauan kemajuan.
4. Deployment, termasuk pembuatan versi, penskalaan, penyediaan dan control akses.

Selain itu pipeline membantu pada proses mediapipe dengan dapat mendeteksi model telapak tangan yang beroperasi dengan gambar penuh dan mengembalikan kotak pembatas tangan yang berorientasi. Model tangan yang beroperasi pada area gambar dapat dipangkas dan ditentukan oleh detector telapak tangan sehingga akan mengembalikan titik kunci tangan 3D dengan fidelitas tinggi.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Pipeline diimplementasikan sebagai grafik mediapipe yang menggunakan subgraph pelacakan tangan dan modul tangan, serta dirender dengan subgraph perender tangan secara khusus.

2.2.6 Palm Detector

Palm detector berfungsi sebagai pendeteksi lokasi awal pada tangan,dengan menggunakan single shot detector yang dioptimalkan secara real time, dan juga diimplementasikan untuk mediapipe (Zhang et al., 2020). Mendeteksi tangan merupakan tugas yang kompleks, oleh sebab itu model harus mampu beradaptasi untuk berbagai ukuran. Sebagai solusi dari cara untuk mengantisipasi kegagagln maka Langkah pertama melatih detector telapak tangan, bukan detector tantan dikarenakan dengan begitu dapat memperkirakan kotak pembatas benda atau telapak tangan. Kotak pembatas berbentuk persegi. Kedua dengan menggunakan ekstraktor atau fitur encode yang mirip dengan FPN dengan model dapat beradaptasi untuk bagian lebih besar maupun objek kecil.

2.2.7 Hand Landmark Model Menggunakan *Mediapipe*

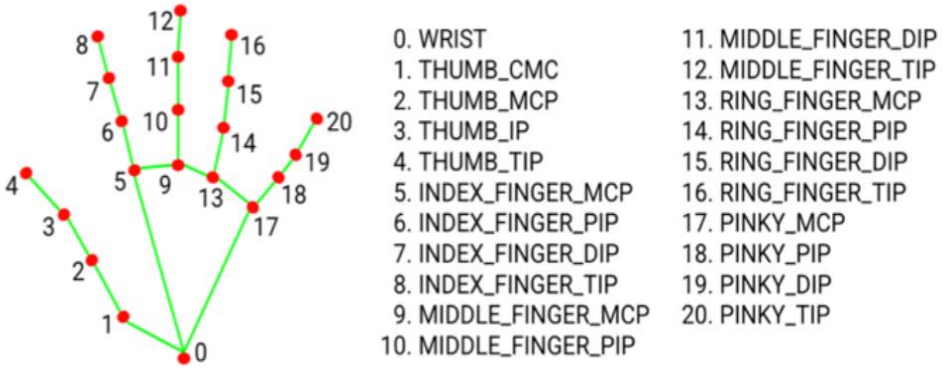
MediaPipe adalah kerangka kerja yang memungkinkan pengembang untuk membangun saluran ML multi-modal (video, audio, seri waktu apa pun) yang diterapkan lintas platform. MediaPipe memiliki banyak koleksi model deteksi dan pelacakan tubuh manusia yang dilatih pada kumpulan data Google yang sangat besar dan paling beragam. Sebagai kerangka node dan tepi atau landmark, mereka melacak titik-titik kunci pada bagian tubuh yang berbeda. Semua titik koordinat dinormalisasi tiga dimensi.

Model yang dibuat oleh developer Google menggunakan Tensorflow lite memfasilitasi aliran informasi yang mudah disesuaikan dan dimodifikasi melalui grafik. MediaPipe terdiri dari node pada grafik yang umumnya ditentukan dalam file pbtxt. Node ini terhubung ke file C++. Perluasan pada file-file adalah kelas kalkulator dasar di Mediapipe. Sama seperti aliran video, kelas ini mendapatkan kontrak aliran media dari node lain dalam grafik dan memastikan bahwa itu terhubung. Setelah node-node pipeline lainnya terhubung, class tersebut akan menghasilkan output data yang diprosesnya sendiri. Objek paket yang merangkum banyak berbagai jenis informasi digunakan untuk mengirim setiap aliran informasi ke setiap kalkulator. Ke dalam grafik, paket samping juga dapat dikenakan, di mana

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

node kalkulator dapat diperkenalkan dengan data tambahan seperti konstanta atau properti statis. Struktur yang disederhanakan dalam alur aliran data ini memungkinkan penambahan atau modifikasi dengan mudah dan aliran data menjadi lebih terkontrol dengan tepat. Solusi pelacakan Tangan memiliki pipeline ML di bagian belakangnya yang terdiri dari dua model yang bekerja secara bergantung satu sama lain: a) Deteksi Telapak Model b) Model Landmark (Halder & Tayade, 2021). Model Deteksi Telapak memberikan gambar telapak tangan yang dipotong secara akurat dan selanjutnya diteruskan ke model landmark.

Proses ini mengurangi penggunaan augmentasi data (yaitu Rotasi, Flipping, Scaling) yang dilakukan dalam model Deep Learning dan mendedikasikan sebagian besar waktunya. kekuatan untuk lokalisasi landmark. Cara tradisional adalah dengan mendeteksi tangan dari bingkai dan kemudian melakukan pelokalan landmark di atas bingkai saat ini. Tapi di Palm Detector ini menggunakan tantangan pipeline ML dengan strategi yang berbeda. Mendeteksi tangan adalah prosedur yang rumit karena Anda harus melakukan gambar pemrosesan dan ambang batas dan bekerja dengan berbagai ukuran tangan yang mengarah pada konsumsi waktu. Alih-alih langsung mendeteksi tangan dari arus bingkai, pertama, detektor Palm dilatih yang memperkirakan kotak pembatas di sekitar benda kaku seperti telapak tangan dan kepalan tangan yang lebih sederhana daripada mendeteksi tangan dengan jari-jari yang digabungkan. Kedua, encoder-decoder digunakan sebagai ekstraktor untuk konteks adegan yang lebih besar



Gambar 2. 4 Hand Landmarks (Halder & Tayade, 2021)

Setelah deteksi telapak tangan dilihat dari seluruh bingkai gambar, model Hand Landmark berikutnya muncul di gambar. Model ini tepat melokalkan 21 koordinat buku jari tangan 3D (yaitu, sumbu x, y, z) di dalam wilayah tangan yang

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

terdeteksi. Modelnya sangat terlatih dan kuat dalam pendeteksian tangan sehingga memetakan koordinat ke tangan yang terlihat sebagian. Gambar 2.5 menunjukkan deteksi 21 titik landmark dengan model Hand Landmark. Sekarang kami memiliki model deteksi Telapak Tangan dan Tangan yang berfungsi, model ini melewati kumpulan data kami dari berbagai bahasa. Mengingat Dataset Bahasa Isyarat Amerika, kami memiliki huruf a sampai z. Jadi, kami melewati model deteksi kami di setiap folder alfabet yang berisi gambar dan melakukan deteksi tangan yang menghasilkan 21 titik landmark seperti yang ditunjukkan pada Gambar 2.5. Titik landmark yang diperoleh kemudian disimpan dalam file format CSV. Secara simultan, tugas eliminasi dilakukan saat mengekstrak titik tengara. Di sini, hanya koordinat x dan y yang terdeteksi oleh Hand Landmark model dipertimbangkan untuk melatih model ML. Tergantung pada ukuran dataset sekitar 10-15 menit diperlukan untuk ekstraksi Landmark.

Penelitian (Zhang et al., 2020) dengan judul "MediaPipe Hands: On-device Real-time Hand Tracking" untuk mendapatkan data yang benar, maka perlu dikelompokkan dengan sudut pandang aspek masalah sebagai berikut:

1. Kumpulan data acak, kumpulan data ini berisi 6K gambar dengan berbagai macam, kondisi serta pencahayaan dan tampilan tangan keterbatasan dataset ini tidak mengandung kompleks artikulasi tangan.
2. Kumpulan data gerakan yang dikumpulkan secara internal: Kumpulan data ini berisi 10K gambar yang mencakup berbagai sudut dari semua gerakan tangan secara bentuk fisik. Batasan ini terdapat kumpulan datanya hanya 30 orang dengan latar belakang yang terbatas. Pada alam liar dan dataset internal adalah pelengkap yang bagus satu sama lain untuk meningkatkan ketangguhan.
3. Dataset sintetis: untuk menutupi kemungkinan pose tangan dengan lebih baik dan memberikan pengawasan tambahan untuk kedalaman, dengan membuat model tangan sintetis berkualitas tinggi atas berbagai latar belakang dan peta ke koordinat 3D yang sesuai. Penelitain dengan memakai 3D komersial model tangan yang dipasang dengan 24 tulang dan termasuk 36 blendshapes, yang mengontrol ketebalan jari dan telapak tangan. Modelnya juga menyediakan 5 tekstur dengan warna kulit yang berbeda. Dengan

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

membuat video transformasi antara pose tangan dan mengambil sampel 100 ribu gambar dan video. Dengan setiap pose memiliki lingkungan pencahayaan rentang dinamis tinggi acak dan tiga kamera berbeda. Dataset ini dapat terlihat seperti Gambar berikut ini:

Gambar 2. 5 Data Sintetis (Zhang et al., 2020)

Untuk pendeteksi telapak tangan, hanya menggunakan dataset in-the-wild, yang cukup untuk melokalisasi tangan dan menawarkan variasi penampilan tertinggi. Namun, semua dataset digunakan untuk melatih model tengara tangan. Membubuhi keterangan gambar dunia nyata dengan 21 landmark dan menggunakan sambungan 3D groundtruth yang diproyeksikan untuk gambar sintetis. Untuk kehadiran tangan, dipilih subset gambar dunia nyata sebagai contoh positif dan sampel di wilayah tidak termasuk wilayah tangan beranotasi sebagai contoh negatif. Untuk wenangan, dibubuhkan keterangan subset gambar dunia nyata dengan tangan untuk memberikan data tersebut.

2.2.8 *Open Source Computer Vision Library*

Open Source Computer Vision Library atau biasa dikenal dengan OpenCV adalah API Library dengan menyediakan data recognition dengan memakai bahasa program python. OpenCV mampu menyediakan berbagai dokumentasi untuk beberapa platform seperti ios, android, dan website. OpenCV resmi diluncurkan pada tahun 1999 oleh Intel Research digunakan untuk memajukan aplikasi intensif CPU, bagian dari serangkaian proyek termasuk dalam penelusuran ray tracing dan display 3D.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Pustaka OpenCV merupakan kumpulan dari algoritma dan fungsi C/C++ dari beberapa kelas yang mengimplementasikan beberapa pemrosesan gambar serta algoritma dari visi computer. Pada pengembangannya aktif di antarmuka dengan bahasa Python, ruby, matlab dan bahasa lainnya. OpenCV ditulis dalam fungsi C dengan mengopimalkan serta memanfaatkan prosesor multicore. OpenCV secara otomatis menggunakan *Intel Integrated Primitives* (IPP) yang sesuai saat runtime jika Pustaka tersebut diinstal. OpenCV berisi lebih dari 500 fungsi yang dapat menjangkau banyak area dalam penglihatan, termasuk inspeksi produk pabrik, pencitraan medis, keamanan, antarmuka pengguna, kalibrasi kamera, penglihatan stereo, dan robotika.

OpenCV menggunakan DirectX, yang merupakan kumpulan API yang telah dikembangkan oleh Microsoft untuk membuat aplikasi dan game multimedia. Salah satu tujuan adanya OpenCV adalah menyediakan infrastruktur visi computer yang mudah digunakan untuk membantu orang dengan membangun aplikasi visi yang cukup canggih serta cepat. Berikut beberapa tujuan lainnya adanya OpenCV (Mohamad, 2015):

a. Tingkatkan penelitian visi dengan menyediakan tidak hanya kode terbuka tetapi juga dioptimalkan untuk dasar infrastruktur visi.
b. Menyebarkan pengetahuan visi dengan menyediakan infrastruktur umum yang pengembang dapat dibangun, sehingga kode akan lebih mudah dibaca dan ditransfer.
c. Aplikasi komersial berbasis visi lanjutan dengan membuat kode portabel yang dioptimalkan kinerja tersedia secara gratis dengan lisensi yang tidak memerlukan komersial aplikasi untuk dibuka atau dibebaskan sendiri.

2.2.9 Python

Python merupakan bahasa pemrograman dengan memiliki focus terhadap tingkat keterbacaan kode. Tidak hanya itu Python juga mendapat klaim sebagaimana bahasa dengan pengabungan kemampuan, kapabilitas, memiliki sintaks kode sangat jelas, serta mempunyai fungsionalitas yang komprehensif dan Pustaka standar yang besar (Syahrudin & Kurniawan, 2018).

Python mampu beradaptasi disegala bahasa pemrogaman ataupun paradigma pemrograman. Python memiliki fitur yang sudah dilengkapi dengan

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

adanya manajemen memori otomatis sebagai bahasa pemrograman dinamis. Sama seperti bahasa pmrograman dinamis lain, python digunakan untuk bahasa script walaupun pada praktinya pemakaiaan bahasa python pada pemanfaatan umum tidak menggunakan bahasa script. Python bisa dipakai untuk berbagai pengembangan software dan mampu berjalan diseluruh system operasi. Seperti Linux, Mac OS X, Amiga, Java Virtual Machine, Palm, Windows dan lainnya.

Python mampu didistribusikan dari dan dengan beberapa lisensi yang berbeda ataupun versi yang berbeda. Dikarenakan pada hakikatnya python dapat diperoleh dan digunakan secara bebas, bahkan untuk kebutuhan komersial. Lisensi python tidak bertentangan menurut GPL ataupun open source.

2.2.10 Integrasi Keilmuan

Menjadi lulusan mahasiswa UIN Sunan Ampel Surabaya tentu perlu adanya keterkaitan kajian keilmuan dengan dasar landasan agama Islam. Pada penelitian ini untuk mendapatkan infromasi mengenai korelasi dengan kitab suci agama Islam yakni Al-Qur'an, oleh sebab itu peneliti bermaksud melakukan wawancara dengan Bapak Dr. H. M. Shodiq, S.Ag, M.Si sebagai pengasuh Yayasan Wak Kaji Shodiq dan juga dosen prodi Sosiologi di UIN Sunan Ampel Surabaya pada tanggal 05 November 2021 dengan dapat berkesempatan wawancara langsung digedung Rektorat Lama UIN Sunan Ampel Surabaya.

Perkembangan teknologi menjadikan manusia untuk terus belajar dan berkembang, untuk dapat membaca dan berkembang perlunya untuk memahami keadaan sekitar sehingga sebagaimana firman Allah pada Al Qur'an Surat Al-Alaq Ayat 1 yang merupakan kali pertama ayat suci al quran turun ke dunia sebagai berikut :

اِقْرَاْ بِاسْمِ رَبِّكَ الَّذِيْ خَلَقَۚ

Artinya: Artinya: "Bacalah dengan (menyebut) nama Tuhanmu yang menciptakan,"

ayat tersebut menegaskan bahwasannya Islam sangat memperhatikan keadaan lingkungan sekitar dengan terus membaca keadaan yang sering terjadi perubahan sehingga perlunya teknik analisa yang baik sehingga nantinya akan membawa dampak positif bagi sesama makhluk hidup terutama hubungan manusia dengan

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

makhluk sosial.

Dan juga dengan adanya perubahan manusia setiap harinya maka akan mendatangkan kesulitan yang berbeda pula setiap waktu hal ini berikaitan dengan firman Allah pada surat al-insyirah ayat 5 yaitu:

فَاِنَّ مَعَ الْعُسْرِ يُسْرًا

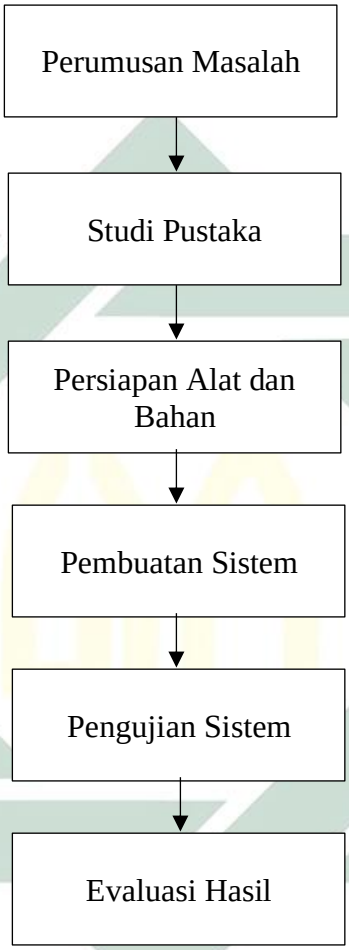
Artinya : Maka sesungguhnya beserta kesulitan ada kemudahan,

Maksud dari kedua ayat tersebut adalah manusia harus pandai membaca keadaan lingkungan sekitar sehingga saling peka dan dapat bermanfaat dengan baik secara seksama makhluk sosial sehingga manusia sebisa mungkin selalu hadir sebagai pemecah masalah dalam kehidupan sehari hari terutama untuk seluruh dunia sebagai agen perubahan yang baik dan bersyariat.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

BAB III
METODOLOGI PENELITIAN

Alur penelitian *hand gesture recognition* dengan web servis untuk *pointing device* sebagai piranti masukan dapat digambarkan pada Gambar 3.1 Berikut ini:



Gambar 3. 1 Alur Penelitian

3.1 Perumusan Masalah

Pada tahap ini ditemukan permasalahan mengenai tidak berkembangnya piranti masukan pada komputer. Sehingga ketika dipakai belum mencapai standart persepsi bahwa pengguna akan nyaman memakai piranti komputer termasuk inputnya saat mereka merasa hubungan interaksi manusia dan komputer berjalan sangat alami seperti saat menjalani kehidupan sehari-hari dengan menggunakan gerakan tubuh untuk berkomunikasi terutama gerakan tangan. Sehingga muncul ide untuk

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

mengagas perkembangan piranti masukan dengan memanfaatkan *Hand Landmark Model*, dengan mekanisme hanya dengan gerakan telapak tangan maka mouse dapat tergantikan. Menggunakan teknologi SSD untuk mendeteksi citra dari tangan dan memberikan marker menggunakan Hand Landmark dan Palm Detector. Dan dalam riset ini mengambil referensi dari riset sebelumnya, memiliki kendala pada permasalahan pencahayaan dan juga terdapat bentuk yang berbeda.

3.2 Studi Pustaka

Dalam penelitian ini tentu membutuhkan beberapa referensi dari beberapa penelitian terdahulu dengan bidang yang sama. Maka untuk referensi akan diambil dari beberapa jurnal terpercaya yang berisi berbagai penelitian berkaitan dengan pendeteksian tangan. Serta beberapa aplikasi terdahulu juga digunakan sebagai bahan rujukan dalam melakukan penelitian berkaitan dengan hand gesture recognition.

3.3 Persiapan Alat dan Bahan

Untuk beberapa alat dan bahan yang dibutuhkan dalam penelitian ini adalah:

1. PC/Komputer/Labtop

Spesifikasi *Hardware*:

• Ram 8 Gb

• VRam 1 Gb

• Processor speed 2 Ghz

• Kamera berukuran 720 x 480 *pixel* (HD)

Spesifikasi *Software*:

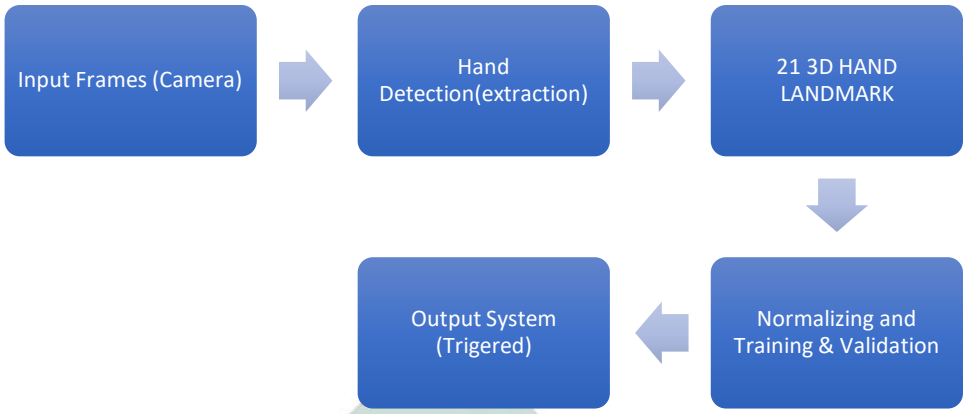
• Terinstal Python

• Sistem operasi windows 7, 8, dan 10

3.4 Pembuatan Sistem

Setelah mempersiapkan alat dan bahan yang didapat melalui berbagai referensi, tahap selanjutnya adalah mengenai implementasi dari alur sistem *Hand Gesture Recognition.*

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id



Gambar 3. 2 Alur Sistem Hand Gesture Recognition berbasis Hand Landmark Model

Secara singkat cara kerja dari software ini adalah dengan memakai media kamera dari laptop/pc, kamera berfungsi sebagai identifikasi telapak tangan yang nantinya akan diekstraksi menggunakan metode SSD menjadi bilangan numerik. Setelah tangan terdeteksi, akan diteruskan menuju *landmark localization* agar secara tepat melokalisasi 21 koordinat buku jari 3D (yaitu, sumbu x, y, z) di dalam wilayah tangan yang terdeteksi. Kemudian diteruskan dalam proses *Normalizing and Training & Validation* yang berfungsi sebagai proses menormalkan koordinat x dan y agar sesuai dengan sistem. file data kemudian disiapkan untuk dipecah menjadi set pelatihan dan validasi. 80% data disimpan untuk melatih model dengan berbagai fungsi pengoptimalan dan kehilangan, sedangkan 20% data dicadangkan untuk memvalidasi model. Lalu dilanjutkan menuju trigger yang sudah dibuat. Selanjutnya, tahapan untuk pembuatan *Hand Gesture Recognition* sebagai berikut:

a. Melakukan Instalasi Python

Pada tahap pertama dilakukan instalansi Python. NodeJS merupakan system perangkat lunak dengan desain pengembangan aplikasi dengan menggunakan bahasa pemrograman JavaScript, mempunyai model event driven (FAJRIN, 2017). Python digunakan karena memiliki kelebihan mudah untuk diaplikasikan pada pengembangan teknologi menurut (Kadarina & Ibnu Fajar, 2019).

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

b. Melakukan Instalasi Package

Pada tahap ini dilakukan instalasi package yang di perlukan untuk kebutuhan system Hand Gesture Recognition berupa OpenCV, Mediapipe dan AutoPY. OpenCV dipakai dengan tujuan khusus sebagai pengolahan citra (Zulkhaidi et al., 2020). Sedangkan mediapipe memiliki fungsi untuk dapat mengenali objek *hand gesture* (Makahaube et al., 2021). AutoPY merupakan Pustaka otomatisasi GUI dengan lintas platform sederhana dengan bahasa python, berfungsi untuk dapat mengontrol keyboard dan mouse, menemukan warna dan bitmap layer, dan akan menampilkan peringatan.

c. Melakukan Pengkodingan Aplikasi

Pada tahap akhir proses pembuatan dilakukan pengkodingan kedalam tools text editor IDLE yang telah disediakan oleh Python dengan memakai versi 3.8 dikarenakan library yang sesuai dengan support, berbeda dengan penggunaan versi 3.9 terdapat kendala pada library tensorflow yang tidak dapat support.

3.5 Pengujian

Pada tahap akhir yaitu pengujian dilakukan setelah pembuatan pengembangan teknologi *hand gesture recognition* selesai dibuat dan bisa dijalankan, yang nantinya akan mengetahui hasil dari penelitian ini.

Selanjutnya dalam melakukan pengujian berdasarkan (Makahaube et al., 2021) melakukan oengujian dengan menggunakan parameter jarak. Parameter jarak ini dilakukan dengan tujuan melihat apakah jarak dapat mempengarui kinerja program atau sistem. Setelah pengujian jarak dilakukan maka selanjutnya akan dihitung dengan menggunakan perhitungan confusion matrix (Halder & Tayade, 2021). Pengujian confusion matrix bertujuan untuk melihat hasil akurasi dari *system hand gesture recognition*. Pengujian terhadap jarak yang dimulai dari 1 meter hingga 2,5 meter. daftar skenario pengujian terhadap jarak terdapat pada Tabel 3.1 berikut:

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Tabel 3. 1 Skenario Pengujian Terhadap Jarak

No.	Skenario Pengujian Jarak	Sub Skenario Pengujian Jarak	Banyak Pengujian	Ekspetasi
1	Tangan dengan jarak 1 meter	Ibu jari (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari telunjuk (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari tengah (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari manis (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari kelingking (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Ibu jari (Kanan)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari telunjuk (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah Klik Kiri
		Jari tengah (Kanan) dan Jari manis (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah double klik
		Jari kelingking (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah klik Kanan
		Sarung Tangan	3x	Dapat terdeteksi
		Sarung Tangan Medis	3x	Dapat terdeteksi
		Cincin	3x	Dapat terdeteksi
2	Tangan dengan jarak 1,5 meter	Ibu jari (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari telunjuk (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari tengah (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

No.	Skenario Pengujian Jarak	Sub Skenario Pengujian Jarak	Banyak Pengujian	Ekspetasi
		Jari manis (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari kelingking (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Ibu jari (Kanan)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari telunjuk (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah Klik Kiri
		Jari tengah (Kanan) dan Jari manis (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah double klik
		Jari kelingking (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah klik Kanan
		Sarung Tangan	3x	Dapat terdeteksi
		Sarung Tangan Medis	3x	Dapat terdeteksi
		Cincin	3x	Dapat terdeteksi
3	Tangan dengan jarak 2 meter	Ibu jari (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari telunjuk (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari tengah (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari manis (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari kelingking (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Ibu jari (Kanan)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari telunjuk (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah Klik Kiri

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

No.	Skenario Pengujian Jarak	Sub Skenario Pengujian Jarak	Banyak Pengujian	Ekspetasi
		Jari tengah (Kanan) dan Jari manis (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah double klik
		Jari kelingking (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah klik Kanan
		Sarung Tangan	3x	Dapat terdeteksi
		Sarung Tangan Medis	3x	Dapat terdeteksi
		Cincin	3x	Dapat terdeteksi
4	Tangan dengan jarak 2,5 meter	Ibu jari (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari telunjuk (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari tengah (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari manis (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari kelingking (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Ibu jari (Kanan)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari telunjuk (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah Klik Kiri
		Jari tengah (Kanan) dan Jari manis (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah double klik
		Jari kelingking (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah klik Kanan
		Sarung Tangan	3x	Dapat terdeteksi
		Sarung Tangan Medis	3x	Dapat terdeteksi
		Cincin	3x	Dapat terdeteksi

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Selanjutnya adalah melakukan pengujian terhadap kondisi pencahayaan pada ruangan menggunakan pengukuran cahaya lux meter. Daftar skenario pengujian terhadap jarak terdapat pada Tabel 3.2 berikut:

Tabel 3. 2 Skenario Pengujian Terhadap Cahaya

No.	Skenario Pengujian Cahaya	Sub Skenario Pengujian Cahaya	Banyak Pengujian	Ekspetasi
1.	Menggunakan cahaya dari 1 lampu 5 watt dalam 1 meter (30 lux)	Ibu jari (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari telunjuk (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari tengah (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari manis (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari kelingking (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Ibu jari (Kanan)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari telunjuk (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah Klik Kiri
		Jari tengah (Kanan) dan Jari manis (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah double klik
		Jari kelingking (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah klik Kanan
		Sarung Tangan	3x	Dapat terdeteksi
		Sarung Tangan Medis	3x	Dapat terdeteksi
		Cincin	3x	Dapat terdeteksi
2.	Menggunakan cahaya dari 2 lampu 5 watt	Ibu jari (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari telunjuk (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

No.	Skenario Pengujian Cahaya	Sub Skenario Pengujian Cahaya	Banyak Pengujian	Ekspetasi
	dalam 1 meter (62 lux)			aktivitas
		Jari tengah (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari manis (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari kelingking (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Ibu jari (Kanan)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari telunjuk (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah Klik Kiri
		Jari tengah (Kanan) dan Jari manis (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah double klik
		Jari kelingking (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah klik Kanan
		Sarung Tangan	3x	Dapat terdeteksi
		Sarung Tangan Medis	3x	Dapat terdeteksi
		Cincin	3x	Dapat terdeteksi
3.	Menggunakan cahaya dari 3 lampu 5 watt dalam 1 meter (82 lux)	Ibu jari (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari telunjuk (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari tengah (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari manis (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari kelingking (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

No.	Skenario Pengujian Cahaya	Sub Skenario Pengujian Cahaya	Banyak Pengujian	Ekspetasi
		Ibu jari (Kanan)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari telunjuk (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah Klik Kiri
		Jari tengah (Kanan) dan Jari manis (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah double klik
		Jari kelingking (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah klik Kanan
		Sarung Tangan	3x	Dapat terdeteksi
		Sarung Tangan Medis	3x	Dapat terdeteksi
		Cincin	3x	Dapat terdeteksi
4.	Menggunakan cahaya dari 4 lampu 5 watt dalam 1 meter (132 lux)	Ibu jari (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari telunjuk (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari tengah (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari manis (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari kelingking (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Ibu jari (Kanan)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari telunjuk (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah Klik Kiri
		Jari tengah (Kanan) dan Jari manis (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah double klik
		Jari kelingking (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah klik

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

No.	Skenario Pengujian Cahaya	Sub Skenario Pengujian Cahaya	Banyak Pengujian	Ekspetasi
				Kanan
		Sarung Tangan	3x	Dapat terdeteksi
		Sarung Tangan Medis	3x	Dapat terdeteksi
		Cincin	3x	Dapat terdeteksi
5.	4 Lampu Ruangan (Neon) dengan daya 144 watt dalam 1 meter (232)	Ibu jari (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari telunjuk (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari tengah (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari manis (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari kelingking (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Ibu jari (Kanan)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari telunjuk (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah Klik Kiri
		Jari tengah (Kanan) dan Jari manis (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah double klik
		Jari kelingking (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah klik Kanan
		Sarung Tangan	3x	Dapat terdeteksi
		Sarung Tangan Medis	3x	Dapat terdeteksi
		Cincin	3x	Dapat terdeteksi
6.	Menggunakan cahaya dari matahari	Ibu jari (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari telunjuk (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

No.	Skenario Pengujian Cahaya	Sub Skenario Pengujian Cahaya	Banyak Pengujian	Ekspetasi
	dalam 1 meter (>2000 lux)			aktivitas
		Jari tengah (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari manis (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari kelingking (Kiri)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Ibu jari (Kanan)	3x	Dapat terdeteksi tapi tidak bisa melakukan aktivitas
		Jari telunjuk (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah Klik Kiri
		Jari tengah (Kanan) dan Jari manis (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah double klik
		Jari kelingking (Kanan)	3x	Dapat terdeteksi dan Melakukan Perintah klik Kanan
		Sarung Tangan	3x	Dapat terdeteksi
		Sarung Tangan Medis	3x	Dapat terdeteksi
		Cincin	3x	Dapat terdeteksi

Berdasarkan skenario pengujian diatas dilakukan pengujian sebanyak 3 kali. Hasil dari pengujian ini berupa waktu kecepatan dan hasil benar atau salah dalam mendeteksi *gesture* tangan, dengan menggunakan persamaan berikut :

$$Akurasi = \frac{Jumlah\ klasifikasi\ benar}{Jumlah\ data\ uji} \times 100\% \qquad (8)$$

$$Kecepatan = \frac{\sum \frac{(\sum \frac{jumlah\ waktu\ pengujian\ sub\ data\ uji}{jumlah\ pengujian\ sub\ data\ uji} x\ 100)}{jumlah\ sub\ data\ uji} x100}{jumlah\ data\ uji} x100 \qquad (9)$$

Seperti pada persamaan (8) dan (9) yang menjelaskan pengukuran akurasi

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

dan kecepatan *virtual mouse* menggunakan *hand gesture recognition.* Selanjutnya untuk menguji tingkat akurasi menggunakan *Recall, Precision,* dan *Accuracy*.

Tabel 3. 3 Confusion Matrix

Predicted Values	Predicted Negative	Predicted Positive
Actual Negative	a	b
Actual Positif	c	d

Keterangan :

a = jumlah prediksi negatif yang benar *(True Negative)*.

b = jumlah prediksi positif yang salah *(False Positive)*.

c = jumlah prediksi negatif yang salah *(False Negative)*.

d = jumlah prediksi positif yang benar *(True Positive)*.

Berdasarkan *confusion matrix* pada table 3.3, *Recall* (r), *Precision* (a) dapat dihitung menggunakan persamaan sebagai berikut :

$$r = \frac{d}{d+c}, p = \frac{d}{d+b} \qquad (10)$$

Dari persamaan di atas dapat diketahui bahwa *Recall* (r) merupakan jumlah prediksi positif yang benar *(True Positive)* dibagi dengan jumlah positif yang sebenarnya dalam penelitian. *Precision* (p) merupakan jumlah prediksi positif yang benar di bagi dengan jumlah prediksi yang diklasifikasikan sebagai positif.

Setelah itu pengujian *Accuracy,* merupakan rasio prediksi yang benar (Positif dan Negatif) dibagi dengan data keseluruhan. Persamaan yang digunakan dalam *Accuracy* sebagai berikut :

$$Accuracy = \frac{d+a}{a+b+c+d} \qquad (11)$$

Persamaan diatas akan digunakan pada skenario pengujian jarak dan cahaya yang akan dijadikan grafik untuk melihat tingkat akurasi pada *Hand Landmark Model*.

Selanjutnya adalah melakukan pengujian fungsionalitasnya untuk mengecek keberhasilan aplikasi dalam menjalanankan fungsi. *Test case scenario*

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

di buat untuk mendapatkan hasil dari fungsi aplikasi, seperti yang terdapat pada table 3.4 berikut:

Tabel 3. 4 Test Case Scenario Pengujian Fungsionalitas Aplikasi

Test Case	***Pre-Condition***	***Expectation Result***
Deteksi tangan dan jari	Tidak ada tangan didepan kamera	Tidak mendeteksi tangan
	1 tangan didepan kamera	Mendeteksi 1 tangan
	2 tangan atau lebih didepan kamera	Tidak dapat mendeteksi 2 tangan atau lebih
	Ada gambar atau pola berbentuk tangan didepan kamera	Tidak mendeteksi tangan
	User memakai sarung tangan	Aplikasi tidak mendeteksi tangan
Klik kiri	Aplikasi mendeteksi jari kelingking (Tangan Kanan)	Mouse klik kanan
	Aplikasi mendeteksi jari kelingking (Tangan Kiri)	Tidak bisa klik kanan
Klik kanan	Aplikasi mendeteksi jari telunjuk (Tangan Kanan)	Mouse klik kiri
	Aplikasi mendeteksi jari telunjuk (Tangan Kiri)	Tidak bisa klik kiri
Double Klik	Aplikasi mendeteksi jari tengah dan jari manis (Tangan Kanan)	Mouse double klik
	Aplikasi mendeteksi jari tengah dan jari manis (Tangan Kiri)	Tidak bisa double klik
Mouse movement	Aplikasi mendeteksi 5 jari (Tangan Kanan)	Mouse bisa digerakkan
	Aplikasi mendeteksi 5 jari (Tangan Kiri)	Mouse tidak bisa digerakkan
	Aplikasi mendeteksi kurang dari 5 jari	Mouse tidak bisa digerakkan

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

3.6 Evaluasi Hasil

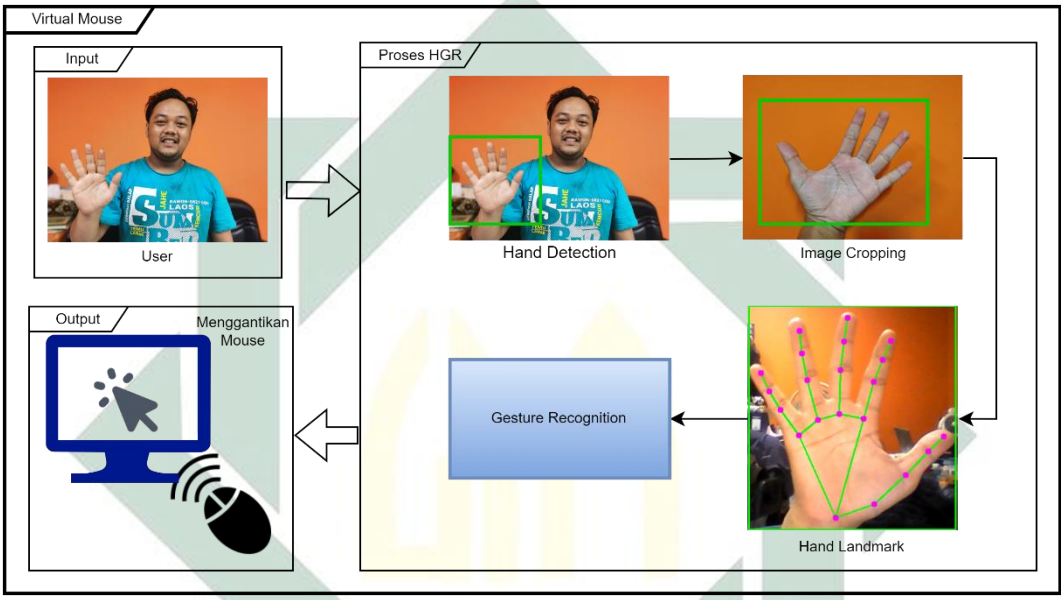
Pada tahap ini semua tahapan sebelumnya akan ditulis dalam sebuah laporan dan akan ditarik kesimpulan dari hasil pengujian aplikasi apa saja yang kekurangan maupun kelebihan dari penggunakan metode *Single Shot Detector* dalam mendeteksi tangan dan *Hand Landmark Model* untuk memberikan marker.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

BAB IV
HASIL DAN PEMBAHASAN

4.1 Desain Arsitektur Sistem

Teknologi Hand Gesture Recognition (HGR) dikebangkan dengan berdasarkan hasil analisa kebutuhan berupa desain arsitektur system, seperti yang ditunjukkan pada Gambar 4.1 Berikut ini:



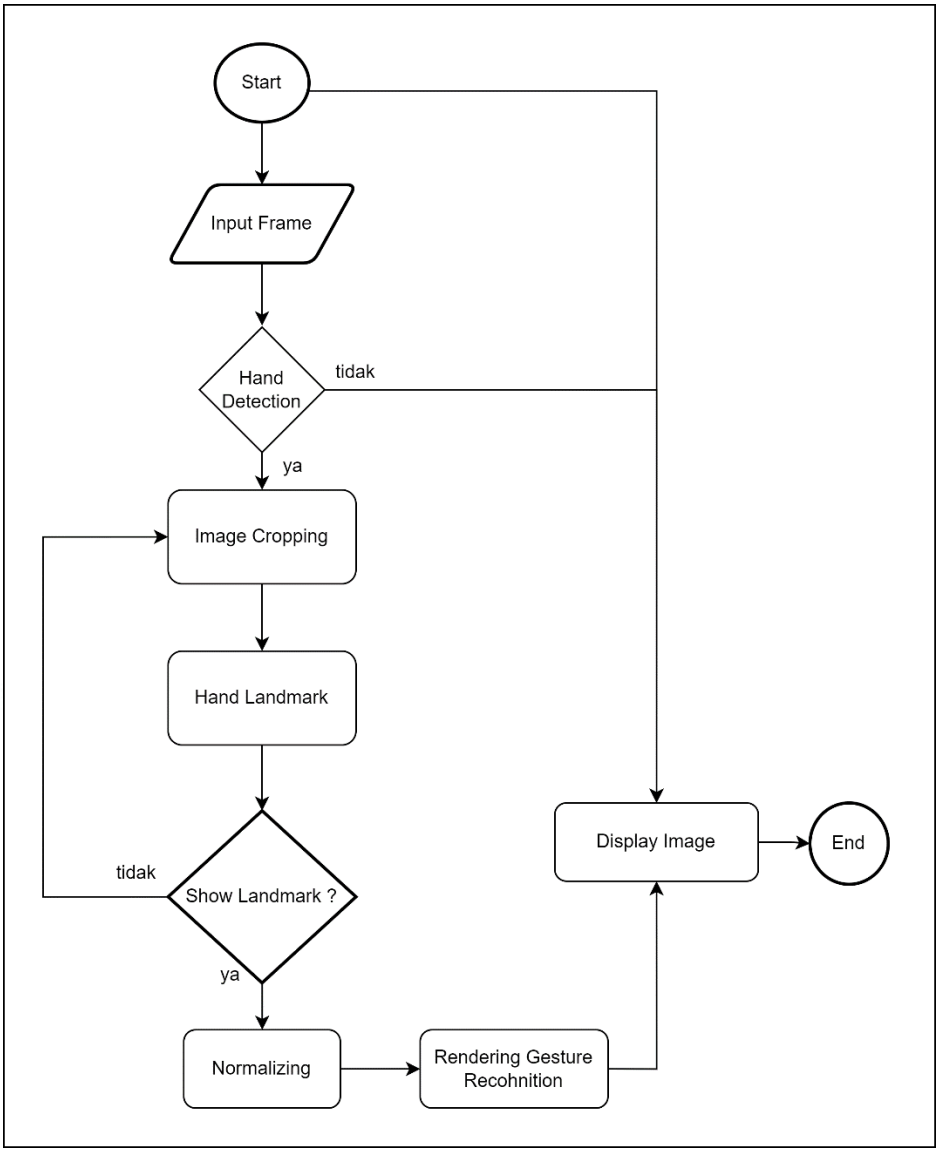
Gambar 4. 1 Desain Arsitektur Sistem

Pada desain arsitektur system terdapat 3 bagian utama yaitu kondisi awal, proses data dan output. Bagian awal untuk mendeteksi image. Setelah image dapat terdeteksi diteruskan pada bagian proses yaitu pertama hand detection untuk mendeteksi tangan, ketika tangan sudah dapat terdeteksi maka hanya akan crop pada bagian tangan saja. Jika pada proses crop telah dijalankan maka proses akan dilanjutkan pada tahap pemberian titik landmark pada tangan. Sedangkan fungsi dari adanya *gesture recognition* adalah untuk membuat fungsi fungsi setiap jari menjalankan perintah mouse yang sudah disesuikan dengan program yang dibuat. Setelah melewati beberapa proses maka Langkah akhir adalah dapat mengeluarkan output yaitu dengan dapatnya penggunaan teknologi HGR yang dimanfaatkan untuk menggantikan mouse sebagai *pointing device*.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

4.2 Alur Penggunaan Aplikasi

Alur penggunaan software dapat dijelaskan pada flowchart yang telah digambarkan pada Gambar 4.2 di bawah ini:



Gambar 4. 2 Alur Penggunaan Aplikasi

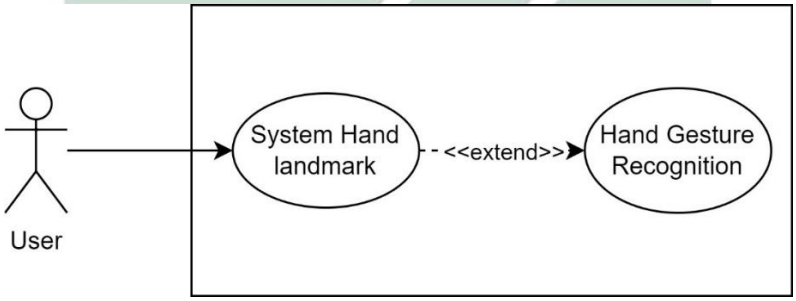
Flowchart dimulai dengan input frame dengan proses detection yang menampilkan gambar secara real time. Pada proses input frame selanjutnya dilakukan pendeteksian, jika dapat terdeteksi maka akan dilanjutkan dengan cropping sehingga mempermudah pada saat proses pemberian landmark pada tangan. Jika tidak dapat terdeteksi maka akan dilanjutkan menuju display image. Selanjutnya, pada proses cropping memotong object gambar terdeteksi yang terdapat pada frame. Setelah proses cropping selesai maka selanjutnya adalah

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

pemberian landmark pada tangan. Pemberian landmark sebanyak 21 titik. Jika proses pemberian landmark berhasil maka proses selanjutnya dengan melakukan normalizing. Jika proses landmark tidak berhasil maka proses Kembali pada image cropping. Selanjutnya dilakukan rendering yang berfungsi untuk memproses output menuju display yang terdapat di layer komputer.

4.3 UML Use Case Diagram

Pada diagram ini, menampilkan skenario dalam penggunaan aplikasi dengan pemakai aplikasi (user). Skenario dari use case menggambarkan pengguna bisa menjalankan fungsi dari aplikasi. Pada aplikasi ini terdapat hanya satu pengguna aplikasi yaitu user. User dapat dikategorikan semua pemakai teknologi terutama computer ataupun laptop. Sesuai dengan hasil analisa user aplikasi bisa menjalankan perangkat pengganti mouse hanya dengan tangan. Sehingga jari-jari tangan berfungsi untuk dapat menggerakan kursor pada layer computer atau laptop. Aplikasi dapat mendeteksi kedua tangan yaitu kiri dan kanan akan tetapi hanya tangan kanan yang dapat menjalankan program aplikasi. Pada saat menjalankan aplikasi, kegiatan user extended terhadap sistem Hand Landmark. Artinya seluruh kegiatan atau fungsi hanya dapat dijalankan ketika proses sistem hand landmark atau fungsi memiliki ketergantungan dengan case hand landmark. Fungsi yang dapat dijalankan user seperti klik kanan, klik kiri, double klik dan drag mouse (kursor) yang berada dalam case Hand Gesture Recognition. Berikut merupakan use case diagram dari aplikasi ini pada Gambar 4.3.



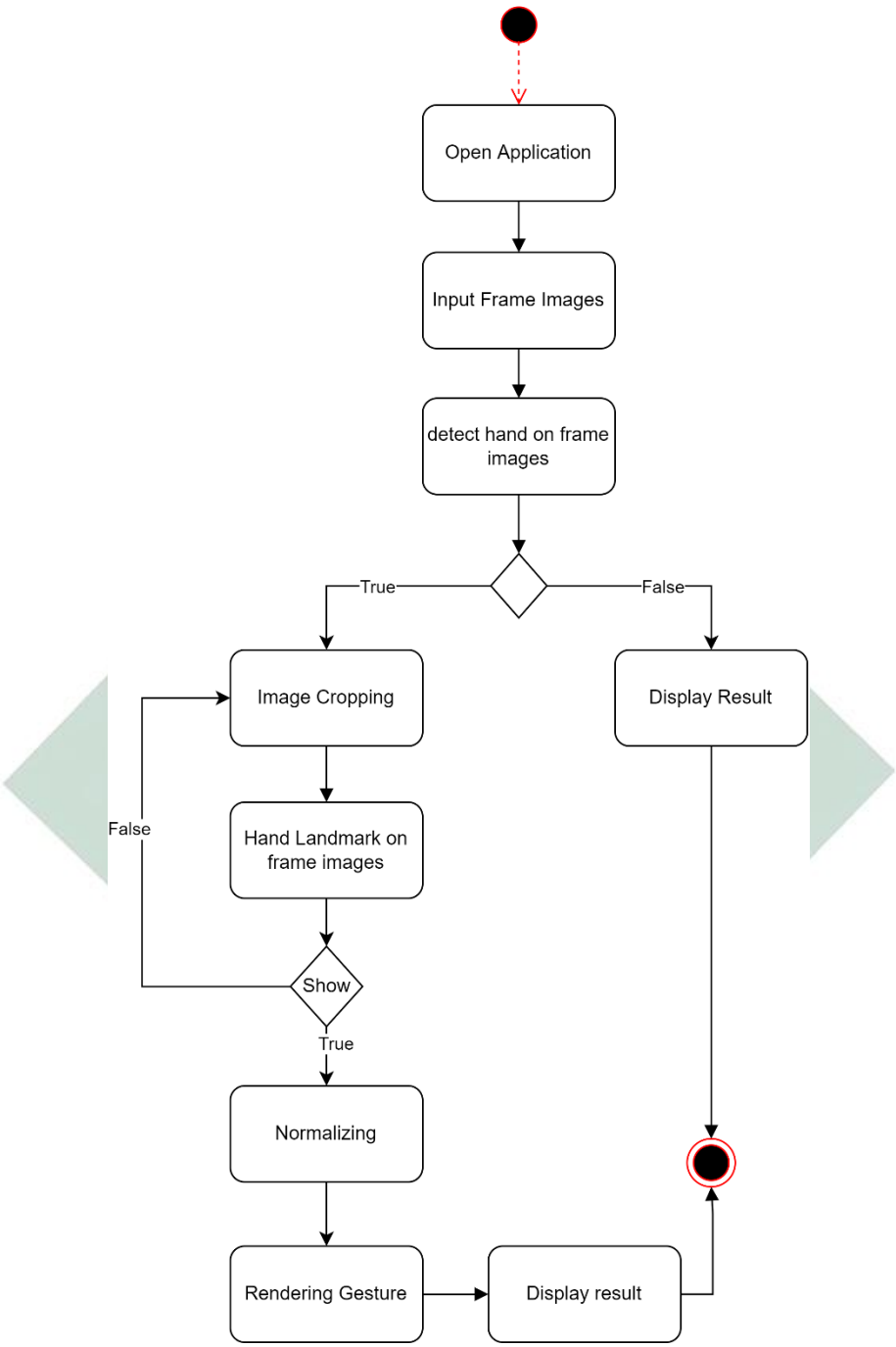
Gambar 4. 3 *Use Case Diagram*

4.4 UML Activity Diagram

Aktivitas system yang dimulai dari awal hingga selesai dapat digambarkan dengan activity diagram. Dengan adanya activity diagram dapat memperlihatkan proses berjalannya aktivitas yang dikerjakan system. Dari empat fungsi task yang tergambarkan pada Gambar Use Case, dibuat menjadi 2 diagram aktivitas sebagai

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

berikut :

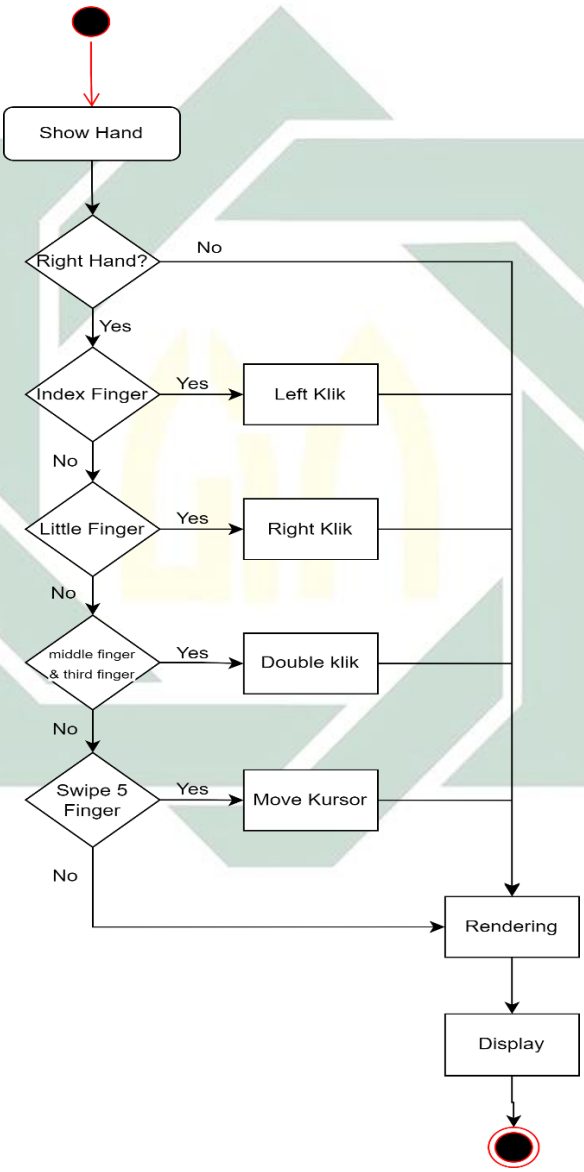


Gambar 4. 4 *UML Activity Diagram Hand Landmark Model*

Dapat dilihat pada Gambar 4.4 Diagram aktivitas dimulai dengan open application sementara otomatis camera menyala, kemudian memasukkan tangan dalam frame. Setelah itu dilakukan decision yaitu jika berhasil untuk dideteksi proses akan berlanjut pada cropping. Jika system tidak dapat melakukan deteksi maka tampilan akan tetap menampilkan display pada layer laptop secara real time.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Setelah system dapat melakukan cropping maka system akan menjalankan proses landmark frame. Setelah itu dilakukan decision show yaitu jika berhasil untuk menambahkan landmark proses akan berlanjut pada normalizing. Jika system tidak dapat menambahkan landmark maka akan kembali ke proses image cropping. Sesudah normalizing dilakukan gesture recognition dan rendering. Jika seluruh proses sudah berjalan dengan baik maka proses penganti mouse dengan tangan dan juga bisa menjalankan kursor maka tampilan akan secara real time menuju display.



Gambar 4. 5 *UML Activity Diagram Gesture Recognition*

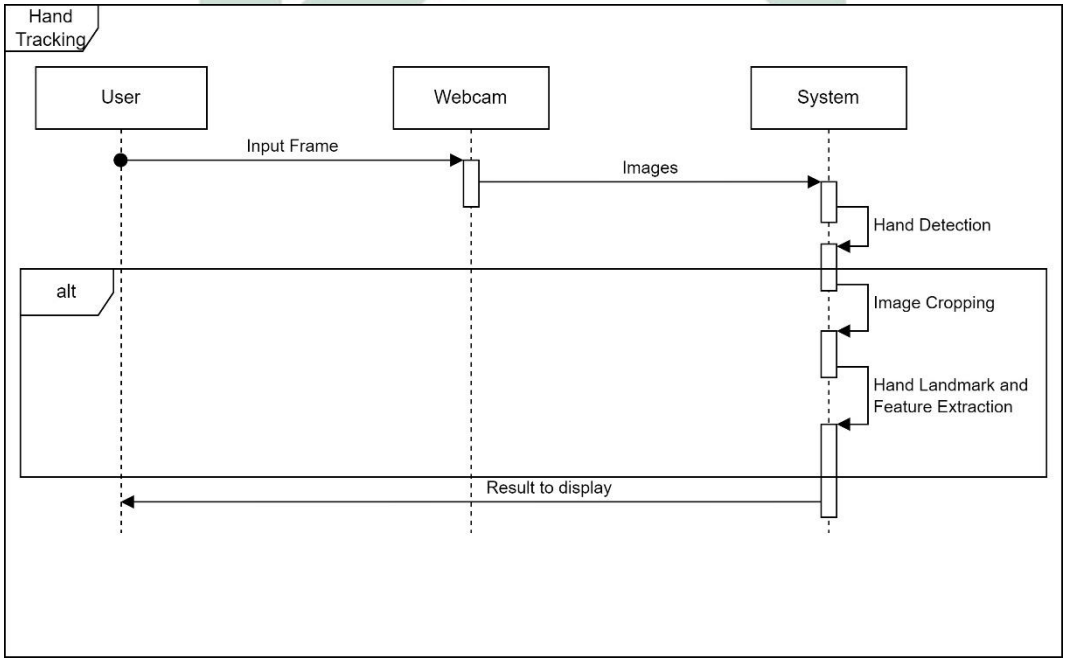
Pada Gambar 4.5 merupakan activity diagram pada proses hand gesture recognition diawali dengan proses melihatkan tangan diverifikasi apakah tangan tersebut merupakan tangan kanan jika tangan tersebut merupakan tangan kanan

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

maka proses selanjutnya dapat di jalankan. Akan tetapi jika tangan tersebut merupakan tangan kiri maka proses akan berhenti dan menampilkan display result. Setelah system dapat mengerti bahwa inputan merupakan tangan kanan maka akan diverifikasi lagi untuk menggerakkan kursor dengan mengunakan 5 jari, jari telunjuk digunakan untuk klik kiri, jari kelingking untuk klik kanan, ibu jari digunakan untuk double klik. Setelah proses selesai maka akan direndering dan ditampilkan pada display.

4.5 UML Squence Diagram

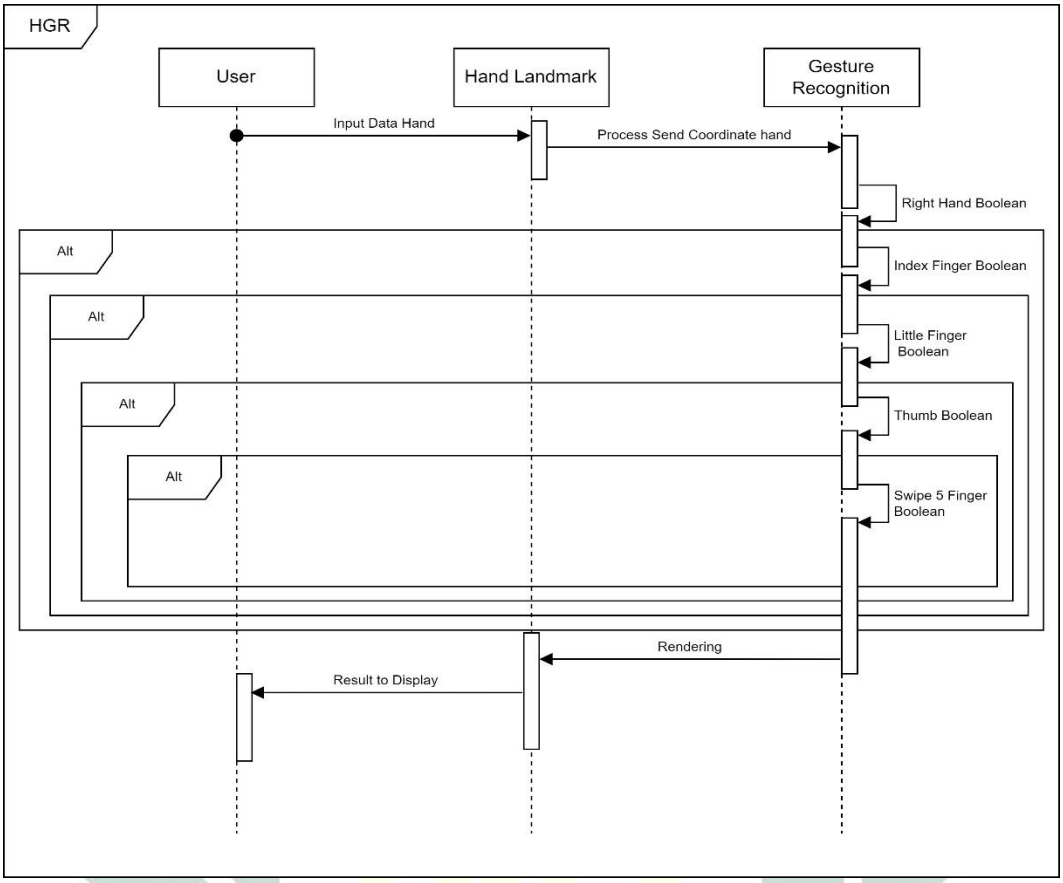
Antar object dapat berinteraksi dengan digambarkan pada squence diagram. Terdapat 4 fungsi utama yang digambarkan pada satu diagram sequence. Sebagai berikut:



Gambar 4. 6 *UML Squence Diagram Hand Landmark Model*

Pada Gambar 4.6 merupakan sequence diagram dengan 3 object yaitu user, webcam dan juga system. User melakukan input frame ke webcam. Setelah itu webcam meneruskan ke system dengan berupa image atau gambar. Gambar oleh system akan dilakukan pengecekan mengenai object tangan. Jika object tangan dapat terdeteksi maka akan diteruskan untuk proses landmark dan mengekstraksi gambar setelah itu akan dilanjutkan dengan proses gesture recognition dan akan menampilkan hasil proses pada display laptop atau computer.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id



Gambar 4. 7 *UML Squence Diagram Gesture Recognition*

Pada Gambar 4.7 merupakan sequence diagram untuk proses hand gesture recognition yaitu mengoperasikan jari tangan untuk menggerakan kursor pada mouse untuk penjelasa lebih detailnya mengenai detail jari yang dapat digunakan untuk menggerakan kursor sama seperti penjelasan pada Gambar Activity Diagram Hand Gesture Recognition.

4.6 Implementasi Sistem

Suatu tahap dimana suatu sistem yang sudah dirancang dioperasikan dalam keadaan sebenarnya disebut tahap implementasi. Selanjutnya, melalui tahap ini dapat diketahui proses sistem dalam bentuk *psedeucode.* Berikut ini merupakan implementasi dari sistem virtual mouse menggunakan hand gesture recognition berbasis hand landmark model untuk pointing device.

4.6.1 *Psedeucode*

1. Import Library

Pertama yang harus dilakukan adalah mengambil library. Ada beberapa library yang dibutuhkan seperti berikut :

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

```
mengambil cv2
mengambil mediapipe as mp
mengambil time
mengambil math
mengambil numpy as np
```

Library pada *psedeucode* diatas adalah :

- Cv2, sebuah *library open source* yang dikembangkan oleh intel yang fokus untuk menyederhanakan programing terkait citra digital.
- Mediapipe, sebuah *library open source* untuk memberikan solusi *Machine Learning* lintas *platform*.
- Time, merupakan *library* waktu.
- Math, sebuah modul matematika yang sudah menjadi standart python tanpa harus install lagi.
- Numpy, atau singkatan dari *Numerical Python* yang berfokus pada *scientific computing*, numpy memiliki kemampuan untuk membentuk objek *N-dimensional array*.

2. Hand Detection

Hand detection digunakan untuk mendeteksi tangan dan membedakan tangan kanan dan kiri. Pendeteksian tangan dimulai dengan pemberian warna pada contour secara otomatis oleh system. Pemberian warna ini berfungsi untuk sistem agar dapat membaca keberadaan tangan dengan memakai imgRGB sebagai berikut:

```
Mendefinisikan fungsi findHands(parameter self, img, draw=True):
memvariabelkan
imgRGB = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
self.results = jalankan proses hand detection dengan parameter imgRGB self.hands.process(imgRGB)
handsType = berisi array []
```

```
Setelah tangan terdeteksi secara baik maka proses akan dilanjutkan kepada proses klasifikasi tangan yaitu tangan kanan dan kiri. Proses klasifikasi tangan dapat dilihat sebagai berikut:
```


digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

```
jika self.results.multi_hand_landmarks != None maka:
jalankan hand in self.results.multi_handedness:
variable handsType= pengambilan label klasifikasi tangan
jalankan handLms in self.results.multi_hand_landmarks:
jika draw benar maka:
jalankan proses pengalokasian landmark
kembalikan nilai img, handsType
```

3. Hand Landmark Model

Pada proses selanjutnya yaitu pemberian landmark pada tangan yang telah terdeteksi. Saat sistem menjalankan proses landmark model menggunakan syntax enumerate untuk menormalizing telapak tangan dengan titik landmark. Telapak tangan menggunakan garis koordinat yang di inisialisasikan sebagai cx dan cy. Proses dapat dilihat sebagai berikut:

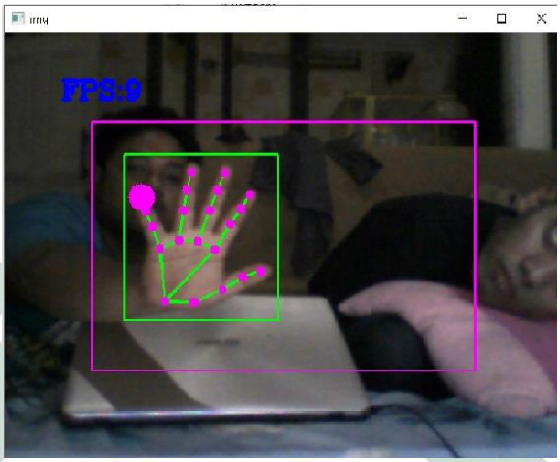
```
Mendefinisikan fungsi findPosition( dengan parameter self,
img, handNo=0, draw=True):
variabel xList = []
variabel yList = []
variabel bbox = []
variabel self.lmList = []
jika self.results.multi_hand_landmarks maka:
variabel myHand = self.results.multi_hand_landmarks[handNo]
jalankan id, lm in menghitung (myHand.landmark):
variabel h, w, c = proses mendapatkan ukurn gambar
variable cx, cy = int(lm.x * w), int(lm.y * h)
menambahkan variabel xList dengan cx
menambahkan variabel yList dengan cy
menambahkan self.lmList.append dengan ([id, cx, cy])
jika draw benar maka:
pembuatan titik titik landmark pada gambar cv2.circle(img,
(cx, cy), 5, (255, 0, 255), cv2.FILLED)
pembuatan bunding box melalui xmin, xmax, ymin, ymax
jika draw benar maka:
cv2.rectangle(img, (xmin - 20, ymin - 20), (xmax + 20, ymax
```


digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

```
+ 20),(0, 255, 0), 2)
mengambalikan nilai self.lmList, bbox
```

4.6.2 Tampilan Sistem *Virtual Mouse*

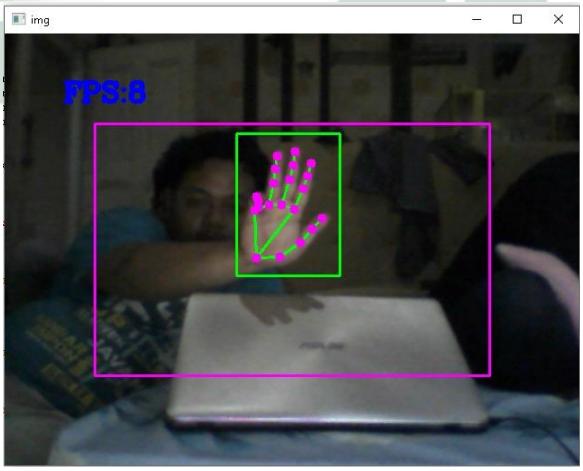
1. Tampilan Aksi *Mouse Move*



Gambar 4. 8 Tampilan Aksi *Mouse Move*

Gambar 4.8 merupakan tampilan dari Hand Gesture Recognition dari inputan 5 jari yang berfungsi sebagai mouse move. Inputan 5 jari dapat terdeteksi ketika inisial ID Finger 4, 8, 12, 16, 20 bernilai 1 atau benar.

2. Tampilan Aksi Klik Kanan



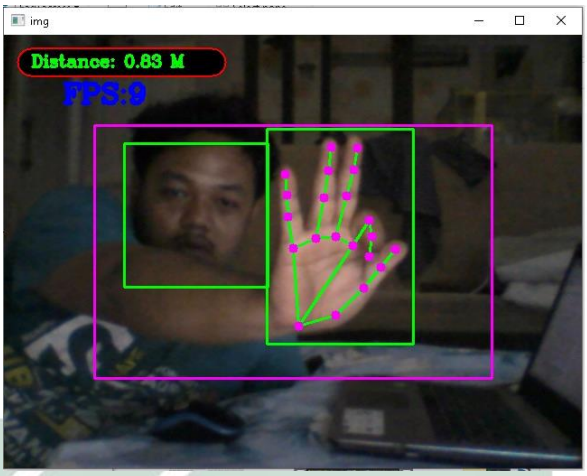
Gambar 4. 9 Tampilan Aksi Klik Kanan

Gambar 4.9 merupakan tampilan dari Hand Gesture Recognition dari inputan jari kelingking turun yang berfungsi sebagai right klik. Inputan jari kelingking dapat terdeteksi ketika inisial ID Finger 4, 8, 12,

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

16 bernilai 1 atau benar dan ID Finger 20 bernilai 0 atau salah.

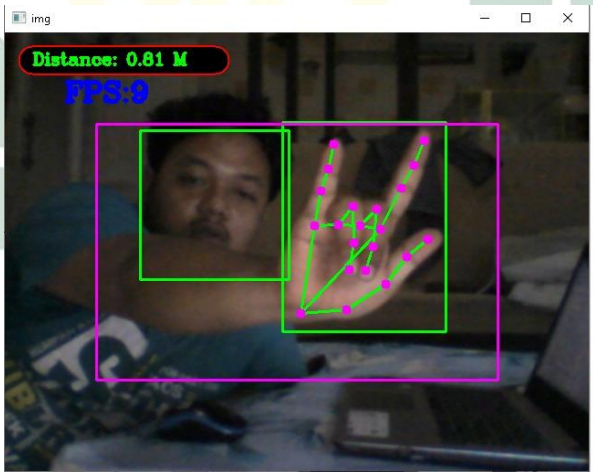
3. Tampilan Aksi Klik Kiri



Gambar 4. 10 Tampilan Aksi Klik Kiri

Gambar 4.10 merupakan tampilan dari Hand Gesture Recognition dari inputan jari telunjuk yang berfungsi sebagai left klik. Inputan jari telunjuk dapat terdeteksi ketika inisial ID Finger 4, 12, 16, 20 bernilai 1 atau benar dan ID Finger 8 bernilai 0 atau salah.

4. Tampilan Aksi Double Klik



Gambar 4. 11 Tampilan Aksi Double Klik

Gambar 4.11 merupakan tampilan dari Hand Gesture Recognition dari inputan jari manis dan jari tengah yang berfungsi sebagai double klik. Inputan jari manis dan jari tengah dapat terdeteksi ketika inisial ID Finger 4, 8, 20 bernilai 1 atau benar dan ID Finger 12, 16 bernilai 0 atau salah.

4.7 Pengujian Fungsionalitas Aplikasi

4.7.1 Skenario pengujian aplikatif dengan pengujian jarak dan cahaya.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Dalam pengujian akan menghasilkan sebuah table yang terdiri dari 7 kolom, yakni s (skenario uji), ss (sub skenario uji), r1 (rata-rata dari 3 kali pengujian setiap sub skenario uji), r2 (rata-rata dari r1), berhasil (apabila 1 berarti pendeteksian benar, dan 0 berarti salah), dan j (jumlah pendeteksian benar). Semua angka yang terdapat pada kolom merupakan waktu yang dibutuhkansistem untuk mendeteksi, apabila ada yang bertulis "-" berarti system tidak dapat mendeteksi. Berikut merupakan hasil dari skenario pengujian pada variable pengujian yaitu jarak dan waktu:

Tabel 4. 1 Hasil Pengujian Pengukuran Jarak

s	**ss**	**waktu (s)**			**r1(s)**	**r2(s)**	**berhasil**			**j**
s1	ss1	0.05	0.04	0.04	0.04333333	0.04583333	1	1	1	3
	ss2	0.04	0.06	0.06	0.05333333		1	1	1	3
	ss3	0.04	0.09	0.04	0.05666667		1	1	1	3
	ss4	0.04	0.04	0.05	0.04333333		1	1	1	3
	ss5	0.04	0.06	0.04	0.04666667		1	1	1	3
	ss6	0.04	0.04	0.04	0.04		1	1	1	3
	ss7	0.04	0.05	0.04	0.04333333		1	1	1	3
	ss8	0.05	0.04	0.04	0.04333333		1	1	1	3
	ss9	0.04	0.05	0.04	0.04333333		1	1	1	3
	ss10	0.04	0.07	0.04	0.05		1	1	1	3
	ss11	0.04	0.04	0.05	0.04333333		1	1	1	3
	ss12	0.05	0.04	0.04	0.04333333		1	1	1	3
s2	ss1	0.05	0.04	0.09	0.06	0.05083333	1	1	1	3
	ss2	0.04	0.04	0.06	0.04666667		1	1	1	3
	ss3	0.04	0,02	0.04	0.04		1	1	1	3
	ss4	0.04	0.07	0.04	0.05		1	1	1	3
	ss5	0.08	0.09	0.05	0.07333333		1	1	1	3
	ss6	0.06	0.04	0.04	0.04666667		1	1	1	3
	ss7	0.04	0.04	0.09	0.05666667		1	1	1	3
	ss8	0.04	0.04	0.04	0.04		1	1	1	3
	ss9	0.04	0.05	0.04	0.04333333		1	1	1	3

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

s	ss	waktu (s)			r1(s)	r2(s)	berhasil			j
	ss10	0.05	0.04	0.05	0.04666667		1	1	1	3
	ss11	0.04	0.06	0.04	0.04666667		1	1	1	3
	ss12	0.09	0.05	0.04	0.06		1	1	1	3
s3	ss1	0.04	0.06	0.04	0.04666667	0.04972222	1	1	1	3
	ss2	0.05	0.04	0.05	0.04666667		1	1	1	3
	ss3	0.04	0.04	0.06	0.04666667		1	1	1	3
	ss4	0.04	0.09	0.05	0.06		1	1	1	3
	ss5	0.04	0.04	0.05	0.04333333		1	1	1	3
	ss6	0.05	0.09	0.07	0.07		1	1	1	3
	ss7	0.04	0.04	0.05	0.04333333		1	1	1	3
	ss8	0.04	0.04	0.06	0.04666667		1	1	1	3
	ss9	0.04	0.04	0.09	0.05666667		1	1	1	3
	ss10	0.04	0.04	0.06	0.04666667		1	1	1	3
	ss11	0.04	0.04	0.04	0.04		1	1	1	3
	ss12	0.04	0.04	0.07	0.05		1	1	1	3
s4	ss1	0.04	0.04	0.04	0.040	0.047	1	1	1	3
	ss2	-	0.04	0.06	0.047		0	1	1	2
	ss3	0.04	0.05	0.04	0.043		1	1	1	3
	ss4	0.04	0.04	0.04	0.040		1	1	1	3
	ss5	-	0.09	0.05	0.047		0	1	1	2
	ss6	0.04	0.07	0.04	0.050		1	1	1	3
	ss7	0.05	0.05	0.06	0.053		1	1	1	3
	ss8	0.09	0.04	0.05	0.060		1	1	1	3
	ss9	0.07	-	0.09	0.043		1	0	1	2
	ss10	0.08	0.04	0.06	0.060		1	1	1	3
	ss11	-	0.04	-	0.000		0	1	0	1
	ss12	0.07	0.09	0.06	0.073		1	1	1	3
Rata-rata (second) = 0.04840278						Akurasi(%) = 96.528	Jumlah = 139			

Berdasarkan Tabel 4.1 diatas merupakan hasil pengujian jarak pada Hand

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Gesture Recognition didapatkan bahwa seluruh pengujian berhasil dengan menapai tingkat akurasi sebesar 96,5% dengan jumlah pengujian sebanyak 144 kali dengan rata-rata kecepatan sistem dapat bekerja 0,048 second. Setelah itu maka sistem perlu diujikan Kembali dalam pengujian dengan berdasarkan cahaya sebagai berikut :

Tabel 4. 2 Hasil Pengujian Berdasarkan Pengukuran Cahaya

s	ss	waktu (s)			r1(s)	r2(s)	berhasil			j
s1	ss1	0.05	0.06	0.1	0.06969	0.056371944	1	1	1	3
	ss2	-	0.06	0.09	0.05		0	1	1	2
	ss3	0.06	0.1	0.08	0.07914		1	1	1	3
	ss4	0.05	-	0.06	0.036666667		1	0	1	2
	ss5	0.09	-	-	0.03		1	0	0	1
	ss6	0.08	0.08	0.07	0.07908		1	1	1	3
	ss7	0.08	0.1	0.05	0.07802		1	1	1	3
	ss8	-	0.06	0.08	0.046666667		0	1	1	2
	ss9	0.1	0.08	0.06	0.07715		1	1	1	3
	ss10	0.06	0.08	0.06	0.06823		1	1	1	3
	ss11	-	-	-	0		0	0	0	0
	ss12	0.08	0.06	0.05	0.06344		1	1	1	3
s2	ss1	0.06	0.05	0.05	0.05669	0.0721	1	1	1	3
	ss2	0.07	0.06	0.08	0.07031		1	1	1	3
	ss3	0.09	0.08	0.06	0.07692		1	1	1	3
	ss4	0.09	0.09	0.08	0.08648		1	1	1	3
	ss5	0.08	0.08	0.06	0.07356		1	1	1	3
	ss6	0.09	0.08	0.09	0.08404		1	1	1	3
	ss7	0.06	0.07	0.08	0.06869		1	1	1	3
	ss8	0.06	0.08	0.07	0.07175		1	1	1	3
	ss9	0.06	0.08	0.08	0.07156		1	1	1	3
	ss10	0.06	0.07	0.06	0.06443		1	1	1	3
	ss11	0.09	0.06	0.07	0.07227		1	1	1	3
	ss12	0.06	0.05	0.09	0.0685		1	1	1	3

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

s	ss	waktu (s)			r1(s)	r2(s)	berhasil			j
s3	ss1	0.07	0.06	0.06	0.06318	0.06348	1	1	1	3
	ss2	0.08	0.08	0.08	0.07702		1	1	1	3
	ss3	0.05	0.08	0.07	0.06639		1	1	1	3
	ss4	0.06	0.06	0.06	0.05995		1	1	1	3
	ss5	0.06	0.07	0.06	0.06338		1	1	1	3
	ss6	0.05	0.05	0.05	0.05218		1	1	1	3
	ss7	0.05	0.07	0.07	0.06252		1	1	1	3
	ss8	0.07	0.07	0.05	0.06236		1	1	1	3
	ss9	0.06	0.06	0.06	0.05793		1	1	1	3
	ss10	0.05	0.08	0.07	0.06559		1	1	1	3
	ss11	0.08	0.06	0.06	0.06848		1	1	1	3
	ss12	0.05	0.06	0.08	0.06272		1	1	1	3
s4	ss1	0.05	0.05	0.05	0.05107	0.05115	1	1	1	3
	ss2	0.06	0.05	0.06	0.0537		1	1	1	3
	ss3	0.05	0.05	0.06	0.05383		1	1	1	3
	ss4	0.04	0.05	0.05	0.05032		1	1	1	3
	ss5	0.05	0.04	0.05	0.04726		1	1	1	3
	ss6	0.05	0.05	0.04	0.04581		1	1	1	3
	ss7	0.05	0.05	0.05	0.05109		1	1	1	3
	ss8	0.05	0.05	0.06	0.05096		1	1	1	3
	ss9	0.05	0.05	0.06	0.05434		1	1	1	3
	ss10	0.04	0.06	0.06	0.05311		1	1	1	3
	ss11	0.06	0.05	0.05	0.05216		1	1	1	3
	ss12	0.05	0.05	0.05	0.05021		1	1	1	3
s5	ss1	0.05	0.05	0.06	0.0542	0.0499	1	1	1	3
	ss2	0.05	0.04	0.05	0.04884		1	1	1	3
	ss3	0.05	0.06	0.05	0.05115		1	1	1	3
	ss4	0.05	0.05	0.04	0.04665		1	1	1	3
	ss5	0.04	0.05	0.05	0.04614		1	1	1	3
	ss6	0.06	0.04	0.04	0.04761		1	1	1	3

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

s	ss	waktu (s)			r1(s)	r2(s)	berhasil			j
	ss7	0.06	0.05	0.05	0.05166		1	1	1	3
	ss8	0.05	0.06	0.05	0.05492		1	1	1	3
	ss9	0.06	0.04	0.05	0.04994		1	1	1	3
	ss10	0.04	0.05	0.05	0.04665		1	1	1	3
	ss11	0.05	0.05	0.06	0.05191		1	1	1	3
	ss12	0.05	0.05	0.05	0.04919		1	1	1	3
S6	ss1	0.04	0.05	0.05	0.04614	0.0451	1	1	1	3
	ss2	0.04	0.04	0.04	0.04235		1	1	1	3
	ss3	0.05	0.04	0.05	0.04603		1	1	1	3
	ss4	0.04	0.05	0.04	0.04377		1	1	1	3
	ss5	0.05	0.04	0.04	0.04424		1	1	1	3
	ss6	0.04	0.04	0.04	0.04302		1	1	1	3
	ss7	0.04	0.05	0.05	0.04685		1	1	1	3
	ss8	0.05	0.05	0.04	0.04526		1	1	1	3
	ss9	0.05	0.05	0.04	0.04673		1	1	1	3
	ss10	0.04	0.04	0.05	0.0443		1	1	1	3
	ss11	0.05	0.05	0.04	0.04634		1	1	1	3
	ss12	0.05	0.04	0.04	0.04588		1	1	1	3
Rata-rata (second) = 0.056350324						Akurasi(%)= 95%		Jumlah = 207		

Berdasarkan Tabel diatas merupakan hasil pengujian cahaya pada Hand Gesture Recognition didapatkan bahwa seluruh pengujian berhasil dengan menapai tingkat akurasi sebesar 95% dengan jumlah pengujian sebanyak 216 kali dengan rata-rata kecepatan sistem dapat bekerja 0,056 second.

4.7.2 *Recall, Precision*, dan *Accuracy*

a) Pengujian Jarak

Diketahui

TP = 139, TN = 0, FP = 0, FN = 5

- $Recall = \frac{TP}{TP+FN}$

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

$Recall = \frac{139}{139+5}$

$Recall = 0.965278$

$Recall = 96,5\%$

- $Precision = \frac{TP}{TP+FP}$

  $Precision = \frac{139}{139+0}$

  $Precision = 1$

  $Precision = 100\%$

- $Accuracy = \frac{TP+TN}{TP+FP+FN+TN}$

  $Accuracy = \frac{139+0}{139+0+5+0}$

  $Accuracy = 0.965278$

  $Accuracy = 96,5\%$

Dari perhitungan diatas bisa diketahui dengan menggunakan pengujian jarak untuk nilai *Recall* sebesar 96,5%, *Precision* sebesar 100%, *dan Accuracy* sebesar 96,5%.

b) Pengujian Cahaya

Diketahui

TP = 207, TN = 0, FP = 0, FN = 9

- $Recall = \frac{TP}{TP+FN}$

  $Recall = \frac{207}{207+9}$

  $Recall = 0.958$

  $Recall = 95,8\%$

- $Precision = \frac{TP}{TP+FP}$

  $Precision = \frac{207}{207+0}$

  $Precision = 1$

  $Precision = 100\%$

- $Accuracy = \frac{TP+TN}{TP+FP+FN+TN}$

  $Accuracy = \frac{207+0}{207+0+9+0}$

  $Accuracy = 0.958$

  $Accuracy = 95,8\%$

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Dari perhitungan diatas bisa diketahui dengan menggunakan pengujian cahaya untuk nilai *Recall* sebesar 95,8%, *Precision* sebesar 100%, *dan Accuracy* sebesar 95,8%.

4.7.3 *Test Case Scenario Hand Gesture Recognition Virtual Mouse*

Tabel 4. 3 Hasil Pengujian Berdasarkan Test Case Scenario

Test Case	***Pre-Condition***	***Expented Result***	***Real Result***
Deteksi tangan dan jari	Tidak ada tangan didepan kamera	Tidak mendeteksi tangan	True
	1 tangan didepan kamera	Mendeteksi 1 tangan	True
	2 tangan atau lebih didepan kamera	Tidak dapat mendeteksi 2 tangan atau lebih	False
	Ada gambar atau pola berbentuk tangan didepan kamera	Tidak mendeteksi tangan	False
	User memakai sarung tangan	Aplikasi mendeteksi tangan	True
Klik kiri	Aplikasi mendeteksi jari kelingking (Tangan Kanan)	Mouse klik kanan	True
	Aplikasi mendeteksi jari kelingking (Tangan Kiri)	Tidak bisa klik kanan	True
Klik kanan	Aplikasi mendeteksi jari telunjuk (Tangan Kanan)	Mouse klik kiri	True
	Aplikasi mendeteksi jari telunjuk (Tangan Kiri)	Tidak bisa klik kiri	True
Double Klik	Aplikasi mendeteksi jari manis dan jari tengah (Tangan Kanan)	Mouse double klik	True
	Aplikasi mendeteksi jari manis dan jari tengah (Tangan Kiri)	Tidak bisa double klik	True
Mouse	Aplikasi mendeteksi 5 jari	Mouse bisa digerakkan	True

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Test Case	*Pre-Condition*	*Expented Result*	*Real Result*
movement	(Tangan Kanan)		
	Aplikasi mendeteksi 5 jari (Tangan Kiri)	Mouse tidak bisa digerakkan	True
	Aplikasi mendeteksi kurang dari 5 jari	Mouse tidak bisa digerakkan	True

Tabel 4. 4 Hasil Test Case Scenario

	Expected Result	*Actual Result*
Sukses	14	12
Gagal	0	2
Total	14	14
Persentase Kesuksesan	100%	86%
Persentase Kegagalan	0%	14%

Hasil yang ada pada table diatas merupakan hasil pengujian terakhir yang dilakukan pada sistem. Pada pengujian terakhir ini dengan kondisi sebanyak 14 mendapatkan persentase kesuksesan sebesar 86% dan persentase kegagalan 14% berdasarkan dengan *test case scenario* yang ada.

4.8 Evaluasi Sistem

Berdasarkan hasil pengujian yang telah dilakukan mengenai pengujian jarak dan pengujian cahaya didapatkan hasil sebagai berikut :

Tabel 4. 5 Tabel Hasil Evaluasi

Jenis Pengujian	***Recall***	***Precision***	***Accuracy***	**Kecepatan**
Pengujian Jarak	96,5%	100%	96,5%	0.05 Second
Pengujian Cahaya	95,8%	100%	95,8%	0.06 Second

Berdasarkan Tabel 4.4 dapat dikatakan bahwa, setiap aspek pengujian, baik dari pengujian jarak dan pengujian cahaya telah didapatkan hasil diatas rata-rata sebagai berikut. Dalam pengujian jarak dengan jumlah pengujian 144 didapatkan

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

rata-rata hasil kecepatan sistem dalam mendeteksi tangan yaitu 0.05 *Second*, kemudian dari hasil *recall* mendapatkan 96,5%, *Precision* mendapatkan 100%, dan *Accuracy* didapatkan hasil 96,5%. Selanjutnya dalam pengujian cahaya dengan jumlah pengujian 216 didapatkan rata-rata hasil kecepatan sistem dalam mendeteksi tangan yaitu 0.06 *Second* hal ini jauh berbeda dengan hasil penelitian berdasarkan (Ren et al., 2011) yang menghasilkan 0,5 second berbeda sebesar 10x liat lebih cepat, kemudian dari hasil *recall* mendapatkan 95,8%, *Precision* mendapatkan 100%, dan *Accuracy* didapatkan hasil 95,8%. Sehingga dapat dikatakan bahwa sistem yang dibuat sudah memenuhi kriteria *real time* yang berjalan dengan 10x lebih cepat dan sudah berjalan dengan baik.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

BAB V
PENUTUP

5.1 Kesimpulan

Dari hasil penelitian yang dilakukan untuk pembuatan teknologi hand gesture recognition didapatkan kesimpulan sebagaimana berikut ini:

1. Pada pembuatan teknologi hand gesture recognition menggunakan hand detection untuk dapat mendeteksi tangan, hand landmark untuk memberikan point kepada telapak tangan yang sudah terdeteksi agar dapat digunakan untuk menjalankan proses pointing device pengganti mouse, penggunaan telapak tangan dapat di jelaskan secara detail yaitu jari kelingking untuk klik kanan, jari telunjuk klik kiri, jari tengah dan jari manis untuk double klik dan seluruh jari untuk menggerakan kursur atau pointing device mouse.
2. Penerapan *Hand Landmark Model* pada *Virtual Mouse* sebagai *Ponting Device* ,memiliki tingkat akurasi yang tinggi yaitu diatas 90% dan memiliki nilai kegagalan yang rendah yaitu dibawah 10%.
3. Penerapan hand landmark model untuk virtual mouse dapat dikatakan dengan baik dengan bukti Dalam pengujian jarak dengan jumlah pengujian 144 didapatkan rata-rata hasil kecepatan sistem dalam mendeteksi tangan yaitu 0.05 *Second,* kemudian dari hasil *recall* mendapatkan 96,5%, *Precision* mendapatkan 100%, dan *Accuracy* didapatkan hasil 96,5%. Selanjutnya dalam pengujian cahaya dengan jumlah pengujian 216 didapatkan rata-rata hasil kecepatan sistem dalam mendeteksi tangan yaitu 0.06 *Second,* kemudian dari hasil *recall* mendapatkan 95,8%, *Precision* mendapatkan 100%, dan *Accuracy* didapatkan hasil 95,8%. Sehingga dapat dikatakan bahwa sistem menandakan load dapat berjalan dengan baik oleh sistem *Hand Gesture Recognition*, penempatan *Pointing Device* juga sesuai dikarenakan keberhasilan dan kesiapan sistem sudah dapat berjalan dengan baik.

5.2 Saran dan Pemgembangan

Saran untuk pengembangan pada aplikasi ini agar dapat lebih variatif dan dapat digunakan dengan berbagai kalangan yaitu:

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

1. Sistem mampu beradaptasi jika pengguna merupakan seorang disabilitas (tidak memiliki jari)
2. Sistem dapat berjalan dengan menggunakan tangan kanan maupun kiri.
3. Menggunakan metode lain seperti Suport Vector Machine.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Daftar Pustaka

Anwar, S., Sinha, S. K., Vivek, S., & Ashank, V. (2019). Hand gesture recognition: A survey. *Lecture Notes in Electrical Engineering*, *511*(3), 365–371. https://doi.org/10.1007/978-981-13-0776-8_33

Ashari, B. S., Otniel, S. C., & Rianto. (2019). Perbandingan Kinerja K-Means Dengan DSCAN Untuk Metode Clustering Data Penjualan Online Retail. *Jurnal Siliwangi*, *5*(2), 72–77. http://jurnal.unsil.ac.id/index.php/jssainstek/article/view/1283

Defri, A. (2016). *Panduan Pemasangan Opencv Dan Codeblock*. 3–8.

Derakhshan, B., Mahdiraji, A. R., Rabl, T., & Markl, V. (2019). Continuous deployment of machine learning pipelines. *Advances in Database Technology - EDBT*, *2019-March*(April), 397–408. https://doi.org/10.5441/002/edbt.2019.35

FAJRIN, R. (2017). Pengembangan Sistem Informasi Geografis Berbasis Node.JS untuk Pemetaan Mesin dan Tracking Engineer dengan Pemanfaatan Geolocation pada PT IBM Indonesia. *Jurnal Informatika*, *11*(2), 40–47. https://doi.org/10.26555/jifo.v11i2.a6090

Falabiba. (2019). *Modul Interaksi Manusia dan Komputer*.

Garg, P., Aggarwal, N., & Sofat, S. (2009). Vision based hand gesture recognition. *Proceedings of International Conference on Service Science, ICSS*, 260–265. https://doi.org/10.1109/ICSS.2013.40

Halder, A., & Tayade, A. (2021). Sign Language to Text and Speech Translation in Real Time Using Convolutional Neural Network,". *International Journal of Research Publication and Reviews*, *2*, 9–17.

Indrajit, R. E. (2000). Pengantar Konsep Dasar Manajemen Sistem Informasi dan Teknologi Informasi. *STIMIK PERBANAS Renaissance Center*, *10*, 276.

Islam, M. M., Siddiqua, S., & Afnan, J. (2017). Real time Hand Gesture Recognition using different algorithms based on American Sign Language. *2017 IEEE International Conference on Imaging, Vision and Pattern Recognition, IcIVPR*

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

2017. https://doi.org/10.1109/ICIVPR.2017.7890854

K V, D., E, H., Jain D, N., & Reddy B, N. (2020). Hand Gesture Recognition and Voice Conversion for Hearing and Speech Aided Community. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, *6*(3), 223–228. https://doi.org/10.32628/cseit206346

Kadarina, T. M., & Ibnu Fajar, M. H. (2019). Pengenalan Bahasa Pemrograman Python Menggunakan Aplikasi Games Untuk Siswa/I Di Wilayah Kembangan Utara. *Jurnal Abdi Masyarakat (JAM)*, *5*(1), 11. https://doi.org/10.22441/jam.2019.v5.i1.003

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016). SSD: Single shot multibox detector. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *9905 LNCS*, 21–37. https://doi.org/10.1007/978-3-319-46448-0_2

Makahaube, S. S., Sambul, A. M., & Sompie, S. R. U. (2021). *Implementation of Gesture Recognition Technology for Automated Education Service Kiosk*. *16*(4), 465–472.

Masithoh, R. E., Rahardjo, B., Sutiarso, L., & Hardjoko, A. (2012). Pengembangan Computer Vision System Sederhana Untuk Menentukan Kualitas Tomat. *Agritech*, *31*(2), 116–123.

Mohamad, M. (2015). *A Review on OpenCV*. *August*, 1–17. https://doi.org/10.13140/RG.2.1.2269.8721

Munawarah, R., Soesanto, O., & Faisal, M. R. (2016). Penerapan Metode Support Vector Machine. *Kumpulan JurnaL Ilmu Komputer (KLIK)*, *04*(01), 103–113.

Murthy;Jadon. (2009). A Review of Vision Based Dynamic Hand Gestures Recognition. *Computer Science and Application*, *10*(05), 990–1001. https://doi.org/10.12677/csa.2020.105102

Nelson, A., McCombe Waller, S., Robucci, R., Patel, C., & Banerjee, N. (2018). Evaluating touchless capacitive gesture recognition as an assistive device for upper extremity mobility impairment. *Journal of Rehabilitation and Assistive Technologies Engineering*, *5*, 205566831876206. https://doi.org/10.1177/2055668318762063

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Neneng, N., Adi, K., & Isnanto, R. (2016). Support Vector Machine Untuk Klasifikasi Citra Jenis Daging Berdasarkan Tekstur Menggunakan Ekstraksi Ciri Gray Level Co-Occurrence Matrices (GLCM). *Jurnal Sistem Informasi Bisnis*, *6*(1), 1. https://doi.org/10.21456/vol6iss1pp1-10

Ngafifi, M. (2014). Kemajuan Teknologi Dan Pola Hidup Manusia Dalam Perspektif Sosial Budaya. *Jurnal Pembangunan Pendidikan: Fondasi Dan Aplikasi*, *2*(1), 33–47. https://doi.org/10.21831/jppfa.v2i1.2616

Novitasari, N. F., & Saleh, F. (2017). Aplikasi dan persepsi penggunaan teknologi oleh dosen pengajar bahasa inggris di universitas abdurachman saleh situbondo. *Cermin*, *1*(2), 45–53. http://unars.ac.id/ojs/index.php/cermin_unars/article/view/218/171

Ren, Z., Meng, J., & Yuan, J. (2011). *Depth Camera Based Hand Gesture Recognition and its Applications in Human-Computer-Interaction*. 1–5.

Rokade, N. S., Jadhav, H. R., Pathan, S. A., & Annamalai, U. (2016). Hand Gesture Recognition System For Multimedia Applications. *International Research Journal of Engineering and Technology*, 1894–1899. www.irjet.net

Sarkar, A., Patel, K. A., Ram, R. K. G., & Capoor, G. K. (2016). Gesture control of drone using a motion controller. *2016 International Conference on Industrial Informatics and Computer Systems, CIICS 2016*, *October*. https://doi.org/10.1109/ICCSII.2016.7462401

Shi, W. T., Lyu, Z. J., Tang, S. T., Chia, T. L., & Yang, C. Y. (2018). A bionic hand controlled by hand gesture recognition based on surface EMG signals: A preliminary study. *Biocybernetics and Biomedical Engineering*, *38*(1), 126–135. https://doi.org/10.1016/j.bbe.2017.11.001

Simon, T., Joo, H., Matthews, I., & Sheikh, Y. (2017). Hand keypoint detection in single images using multiview bootstrapping. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, *2017-Janua*, 4645–4653. https://doi.org/10.1109/CVPR.2017.494

Sodik, F., Dwi, B., & Kharisudin, I. (2020). Perbandingan Metode Klasifikasi Supervised Learning pada Data Bank Customers Menggunakan Python. *Jurnal Matematika*, *3*, 689–694.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Starner, T., & Pentland, A. (1995). Real-time American Sign Language recognition from video using Hidden Markov models. *Proceedings of the IEEE International Conference on Computer Vision*, 265–270. https://doi.org/10.1109/iscv.1995.477012

Sunyoto, A., & Harjoko, A. (2014). Review Teknik, Teknologi, Metodologi dan Implementasi Pengenalan Gestur Tangan Berbasis Visi. *Seminar Nasional Aplikasi Teknologi Informasi (SNATI)*, H-7. http://jurnal.uii.ac.id/Snati/article/view/3290

Supal, S. N., Hayati, M., & Yatim, M. (2020). Kemahiran meta kognitif kanak-kanak dalam penggunaan perisian dan peranti input tanpa sentuhan leap motion. *Journal of ICT in Education*, *7*(2), 1–13.

Syahrudin, A. N., & Kurniawan, T. (2018). Input Dan Output Pada Bahasa Pemrograman Phyton. *Jurnal Dasar Pemrograman Python STMIK*, *January*, 1–7.

Velaskar, Y., Dulam, A., Sureliya, S., Shenoy, S., & Chouhan, C. (2017). Computer Vision based Hand Gesture Interfaces. *International Journal of Engineering Research & Technology (IJERT)*, *05*(01), 1–4. www.ijert.org

Yuliana, N., Ratri, K., & Wardani, R. (2016). Metode Convex Hull dan Convexity Defects untuk Pengenalan Isyarat Tangan. *Jurnal Telematika*, *11*(2), 81–88.

Yunita, H., & Setyati, E. (2019). Hand Gesture Recognition Sebagai Pengganti Mouse Komputer Menggunakan Kamera. *Jurnal ELTIKOM*, *3*(2), 64–76. https://doi.org/10.31961/eltikom.v3i2.114

Zhang, F., Bazarevsky, V., Vakunov, A., Tkachenka, A., Sung, G., Chang, C.-L., & Grundmann, M. (2020). *MediaPipe Hands: On-device Real-time Hand Tracking*. http://arxiv.org/abs/2006.10214

Zulkarnaen, M. J. (2010). *Efektivitas Pengunaan Teknologi Informasi dan Komunikasi Dalam Meningkatan Prestasi Belajar Pendidikan Agama Islam Siswa kelas X Teknik Komputer dan Informatika di SMK Negeri 3 Bojonegoro*. *c*, 7–22.

Zulkhaidi, T. C. A.-S., Maria, E., & Yulianto, Y. (2020). Pengenalan Pola Bentuk Wajah dengan OpenCV. *Jurnal Rekayasa Teknologi Informasi (JURTI)*, *3*(2), 181. https://doi.org/10.30872/jurti.v3i2.4033

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id