

PROJEK UTS KECERDASAN BUATAN

Virtual Mouse



Dosen Pengampu :

Prof. Dr. Ir. Siti Nurmaini, M.T.

Disusun Oleh :

Arinda Intan Safitri (09011282025041)

SK5A Reguler

FAKULTAS ILMU KOMPUTER

UNIVERSITAS SRIWIJAYA

INDRALAYA

2022

Virtual Mouse

Abstrak

Saat ini teknologi semakin pesat dan sudah menjadi salah satu kebutuhan manusia yang harus dimiliki untuk menyelesaikan permasalahan yang sedang dihadapi. Dalam proyek ini menggunakan 2 metode yaitu Single Shot Detector dan Hand Gesture Recognition menggunakan Framework Mediapipe dan Open Source computer vision system dimana mereka dapat menjalankan fungsi dari masing-masing library. Pada Single Shot Detector (SSD) digunakan untuk menangkap frame kemudian diekstraksi menjadi bilangan numerik. Kemudian Pada pembuatan teknologi hand gesture recognition menggunakan hand detection untuk dapat mendeteksi tangan, hand landmark untuk memberikan titik landmark kepada telapak tangan yang sudah terdeteksi agar dapat digunakan untuk menjalankan proses pointing device pengganti mouse. 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.

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

I. Latar Belakang

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 dihindari. Setiap ide teknologi dihadirkan dengan tujuan memberikan dampak positif bagi kehidupan. Sampai saat ini manusia sudah menikmati banyak manfaat positif dari adanya teknologi. Sehingga teknologi adalah penemuan benda maupun alat yang diwujudkan dari implementasi suatu ilmu pengetahuan serta keterampilan yang dimiliki 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 memberikan perintah. Piranti umum yang biasa dipakai untuk personal computer (PC) adalah keyboard atau mouse. 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 meloncatloncat, 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 . Salah satu solusi adalah memanfaatkan kamera sebagai piranti masukan secara realtime. 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.

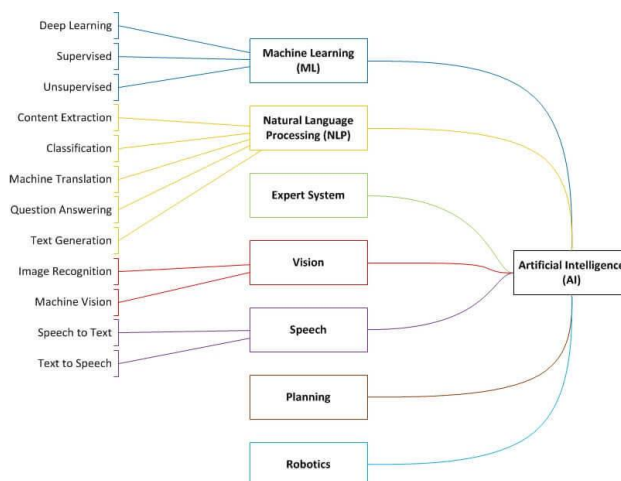
Saat ini pendekatan interaksi antara manusia dan komputer hanya memakai keyboard, mouse, dan pena. Sedangkan hal itu terasa tidak cukup dikarenakan perangkat ini memiliki keterbatasan. Penggunaan camera secara langsung sebagai perangkat masukan adalah metode yang menarik untuk menyediakan Interaksi manusia 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. 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.

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. Kelebihan dengan adanya Hand Landmark lebih mudah diimplementasikan sehingga memiliki output biner, dapat membedakan tangan kanan dan kiri.

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 mengeluarkan trigger. Proyek 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.

II. Dasar Teori

2.1 Machine Learning



Pembelajaran mesin dengan pendekatan artificial intelligence dapat digunakan untuk mengganti dan meniru perilaku manusia untuk pemecahan masalah dilakukan 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 menunjukkan klasifikasi maupun kelompok dengan data berbeda. Model yang dihasilkan dapat berupa model prediksi dengan data yang sudah diberi label. Algoritma yang digunakan untuk Teknik supervised learning diantaranya decision tree, naïve bayes, regresi, K-Nearest Neighbor (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 komputer 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 mempelajari hubungan antar variable target dan fitur. ML memiliki tugas untuk klasifikasi tetapi dalam prosesnya terjadi penentuan class untuk sebuah instance berada yaitu untuk regresi. Regresi ialah memprediksi dari nilai numerik. Regresi termasuk dalam tipe algoritma dengan pembelajaran terarah.

2.2 Computer Vision System (CVS)

Computer Vision System (CVS) merupakan system dengan basis pengembangan algoritma dan juga 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 penerangan 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 detail citra.

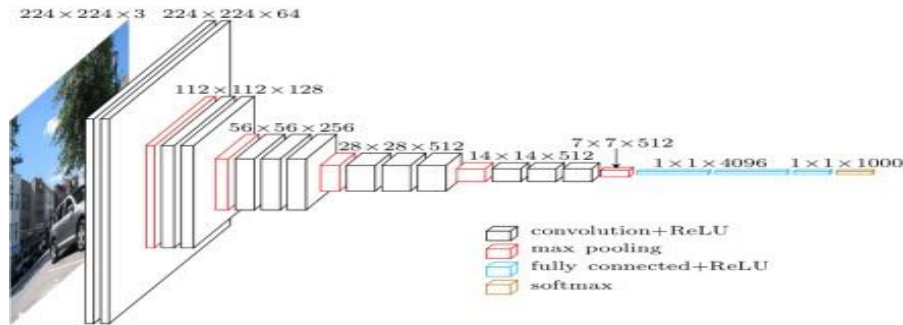
3. Segmentasi citra, dimana citra objek diidentifikasi dan dipisahkan dari latar belakangnya,
4. Pengukuran citra, dimana beberapa fitur-fitur yang sudah diekstrak kemudian dapat diinterpretasikan.

2.3 Single Shot Detector (SSD)

Single Shot Detector (SSD) adalah sebuah metode untuk mengenali atau mendeteksi sebuah *object* pada suatu gambar dengan menggunakan *single deep neural network* dan salah satu algoritma deteksi *object* yang paling populer karena kemudahan implementasi, akurasi yang baik vs rasio yang dibutuhkan komputasi. *Single Shot Detector* (SSD) hanya perlu mengambil satu bidikan tunggal untuk mendeteksi beberapa *object* didalam gambar.

Metode *Single Shot Detector* (SSD) ini termasuk kedalam deteksi *object* secara *real time*, meskipun lebih intuitif daripada rekan-rekannya seperti R-CNN, *Fast R-CNN*, *Faster R-CNN* dan *You Only Look Once* (YOLO), *Single Shot Detector* (SSD) adalah algoritma yang sangat kuat. Menjadi sederhana dalam desain, implementasinya lebih langsung dari GPU dan sudut pandang kerangka kerja pembelajaran yang dalam dan dengan demikian melakukan pengangkatan berat deteksi dengan kecepatan kilat. Juga, poin kunci dari algoritma ini dapat membantu dalam mendapatkan pemahaman yang lebih baik dari metode canggih lainnya.





Arsitektur VGG16

Pada arsitektur SSD termasuk kedalam jenis *Convolutional Neural Network*, *Convolutional Neural Network* (CNN) adalah salah satu jenis *Neural Network* yang biasa digunakan pada data *image*. CNN bisa digunakan untuk mendeteksi dan mengenali *object* pada sebuah *image*.

Arsitektur dari CNN dibagi menjadi 2 bagian besar, *Feature Extraction Layer* dan *Convolutional Layer*. Dimana pada bagian *Feature Extraction Layer* ini adalah melakukan “encoding” dari sebuah *image* menjadi features yang berupa angka-angka yang merepresentasikan *image* tersebut, Sedangkan dibagian *Convolutional Layer* terdiri dari *neuron* yang tersusun sedemikian rupa sehingga membentuk sebuah *filter* dengan panjang dan tinggi (*pixels*). Secara matematis, *Convolutional layer* atau yang dalam Bahasa Indonesianya konvolusi, adalah integral yang mencerminkan jumlah lingkaran dari sebuah sudut fungsi *F* yang digeser atas fungsi *g* sehingga menghasilkan fungsi *h*.

Konvolusi dilambangkan dengan asterisk (*). Sehingga, $F * g = h$ berarti fungsi *F* dikonvolusikan dengan fungsi *g* menghasilkan fungsi *h*. konvolusi dua buah fungsi $F(x)$ dan $g(x)$ di definisikan sebagai berikut:

$$h(x) = F(x) * g(x) = \int F(a)g(x - a)$$

Konvolusi dua buah fungsi $F(x)$ dan $g(x)$ (No 1)

Untuk fungsi diskrit, konvolusi di definisikan sebagai berikut:

$$h(x) = F(x) * g(x) = \sum_{-\infty}^{\infty} F(a)g(x - a)$$

—∞

Konvolusi fungsi diskrit (No 2)

Pada rumus No 3, $g(x)$ disebut dengan kernel konvolusi (*filter*). Kernel $g(x)$ merupakan jendela yang dioperasikan secara bergeser pada sinyal masukan $F(x)$. Hasil konvolusi dinyatakan dengan keluaran $h(x)$. Contoh, misal citra $F(x,y)$ yang berukuran 5x5 sebuah kernel dengan 3x3 matriks sebagai berikut:

$$g(x,y) = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \quad F(x,y) = \begin{bmatrix} 4 & 4 & 3 & 5 & 4 \\ 6 & 6 & 5 & 5 & 2 \\ 5 & 6 & 6 & 6 & 2 \\ 6 & 7 & 5 & 5 & 3 \\ 3 & 5 & 2 & 4 & 4 \end{bmatrix}$$

Citra (x,y) berukuran 5x5 & sebuah kernel dengan 3x3 matriks (No 3)

Secara umum metode *Single Shot Detector (SSD)* mempunyai sebuah rumus sederhana dalam menentukan *default boxes* dan *scale default boxes*, dimana N merupakan jumlah *default boxes*, L_{conf} = *loss classification*, L_{loc} = *loss localization*, L = *prediction box* dan g = *truth ground box*. untuk menentukan *default boxes* bisa di lihat pada Rumus

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

Rumus *Default Boxes* (No 4)

Sedangkan untuk menentukan *scale default boxes* bisa dilihat pada Rumus :

$$S_k = S_{\min} + \frac{S_{\max} - S_{\min}}{m-1} = (k-1), kc[1, m]$$

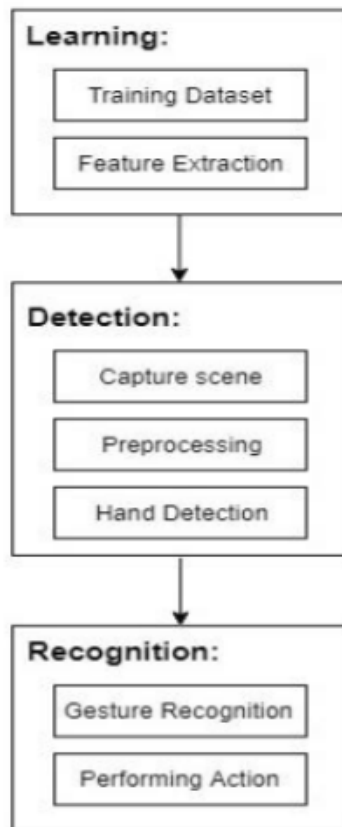
Rumus *Scale Default Boxes* (No 5)

Dimana **smin** adalah lapisan skala terendah, **smax** lapisan skala tertinggi dan **Sk** adalah *input pixels*.

2.4 Hand Gesture Recognition

Hand Gesture Recognition (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.

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 berikut :



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.
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.

2.5 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 subtask yang di enkapsulasi 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.

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

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 . Mendeteksi tangan merupakan tugas yang kompleks, oleh sebab itu model harus mampu beradaptasi untuk berbagai ukuran.

Sebagai solusi dari cara untuk mengantisipasi kegagalan maka Langkah pertama melatih detector telapak tangan, bukan detector tangan 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.

2.7 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 ptxt. 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 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.

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 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.

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.

OpenCV merupakan kumpulan dari algoritma dan fungsi C/C++ dari beberapa kelas yang mengimplementasikan beberapa pemrosesan gambar serta algoritma dari computer vision. Pada pengembangannya aktif di antarmuka dengan bahasa Python, ruby, matlab dan bahasa lainnya. OpenCV ditulis dalam fungsi C dengan mengoptimalkan 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 computer vision yang mudah digunakan untuk membantu orang dengan membangun aplikasi yang cukup canggih serta cepat. Berikut beberapa tujuan lainnya adanya OpenCV :

- a. Tingkatkan penelitian visi dengan menyediakan tidak hanya kode terbuka tetapi juga dioptimalkan untuk dasar infrastruktur computer vision.
- b. Menyebarkan pengetahuan computer vision 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.

2.8 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. Python mampu beradaptasi disegala bahasa pemrograman ataupun paradigma pemrograman. Python memiliki fitur yang sudah dilengkapi dengan adanya manajemen memori otomatis sebagai bahasa pemrograman dinamis. Sama seperti bahasa pemrograman dinamis lain, python digunakan untuk bahasa script walaupun pada praktinya pemakaian 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.

III. Algoritma dan Metode Penerapan

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

```
import cv2
import mediapipe as mp
import pyautogui
import math
from enum import IntEnum
from ctypes import cast, POINTER
from comtypes import CLSCTX_ALL
from pycaw.pycaw import AudioUtilities, IAudioEndpointVolume
from google.protobuf.json_format import MessageToDict
import screen_brightness_control as sbcontrol
```

- **import cv2**
sebuah library open source yang dikembangkan oleh intel yang fokus untuk menyederhanakan programing terkait citra digital.
- **import mediapipe as mp**
sebuah library open source untuk memberikan solusi Machine Learning lintas platform.
- **import pyautogui**
untuk mengotomatisasi pergerakan mouse dan keyboard untuk membangun interaksi dengan aplikasi lain menggunakan skrip Python.
- **import math**
digunakan untuk melakukan operasi matematika
- **from enum import IntEnum**
Dengan bantuan enum. Metode IntEnum(), kita bisa mendapatkan pencacahan berdasarkan nilai bilangan bulat.
- **from ctypes import cast, POINTER**
Fungsi cast () dapat digunakan untuk mentransmisikan instance ctypes ke dalam pointer ke tipe data ctypes yang berbeda. cast () mengambil dua parameter, objek ctypes yang sedang atau dapat dikonversi menjadi pointer dari beberapa jenis, dan tipe pointer ctypes.
- **from comtypes import CLSCTX_ALL**
comtypes memudahkan untuk mengakses dan mengimplementasikan antarmuka COM berbasis kustom dan pengiriman.
- **from pycaw.pycaw import AudioUtilities, IAudioEndpointVolume**

Kita dapat menggunakan library pycaw untuk mengatur volume Audio pada Windows.

- **from google.protobuf.json_format import MessageToDict**

Pengaturan mengontrol pemformatan.

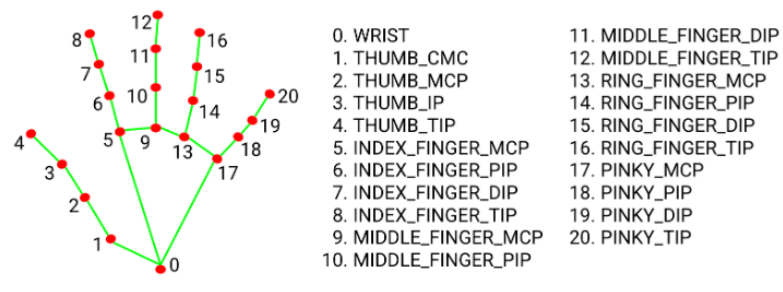
- **import screen_brightness_control as sbcontrol**

untuk mengontrol kecerahan monitor pada Windows.

2. Program melakukan inisialisasi sistem perekaman video menggunakan kamera yang didasarkan pada frame yang telah ditangkap oleh webcam di laptop atau PC, kamera berfungsi sebagai identifikasi telapak tangan yang nantinya akan diekstraksi menggunakan Metode Single Shot Detector (SSD) menjadi bilangan numerik. Setelah tangan terdeteksi dilayar frame WEBCAM, kemudian 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 titik landmark yang diperoleh kemudian disimpan dalam file berformat CSV. Lalu sistem menjalankan proses landmark model menggunakan syntax enumerate untuk diteruskan dalam proses Normalizing and Training & Validation yang berfungsi sebagai proses menormalkan koordinat x dan y agar sesuai dengan sistem file data. Jika proses landmark tidak berhasil maka proses Kembali pada image cropping. Selanjutnya dilakukan rendering yang berfungsi untuk memproses output menuju display yang terdapat di layar komputer.

3. Secara simultan, tugas eliminasi dilakukan saat mengekstrak titik landmark. Di sini, hanya koordinat x dan y yang terdeteksi oleh Hand Landmark model dipertimbangkan untuk melatih model Machine Learning. Tergantung pada ukuran dataset sekitar 10-15 menit diperlukan untuk ekstraksi Landmark. kemudian disiapkan untuk dipecah menjadi set Trainning 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.

Berikut ini merupakan titik landmark yang terdiri dari 21 titik :



Kemudian pengambilan objek video dibuat menggunakan perintah `imshow` atau `imread` `divideo capture(0)`. Mengapa 0 ? karena 0 merupakan indeks pertama yang berarti kamera driver utama yang digunakan untuk read. Setelah itu kamera web akan mulai merekam video.