

# Hand KeyPoint Detection menggunakan Algoritma Single Shot Detector (SSD)

Mursyidah<sup>1\*</sup>, Ilham Safar<sup>2</sup>, Herri Mahyar<sup>3</sup>

<sup>1,2</sup> Program Studi Teknologi Rekayasa Mutimedia, Jurusan Teknologi Informasi dan Komputer, Politeknik Negeri Lhokseumawe

<sup>3</sup> Program Studi Teknologi Rekayasa Konstruksi Bangunan Gedung, Politeknik Negeri Lhokseumawe  
e-mail: [mursyidah@pnl.ac.id](mailto:mursyidah@pnl.ac.id), [ilhamsafar@pnl.ac.id](mailto:ilhamsafar@pnl.ac.id), [herrimahyar@pnl.ac.id](mailto:herrimahyar@pnl.ac.id)

**Abstrak**— Keyboard merupakan salah satu piranti masukan komputer yang terdiri dari huruf, angka, dan fungsi kontrol lainnya. Beberapa tugas muncul dalam bentuk teks yang diinputkan ke dalam sistem komputer melalui keyboard. Dalam penggunaannya keyboard terdapat permasalahan yang sering terjadi misalnya tombol keyboard yang tidak berfungsi, selain itu juga keyboard dapat mengetik sendiri tanpa diperintah yang diakibatkan oleh tombol keyboard yang sudah tenggelam. Sehingga penggunaan keyboard menjadi kurang efisien. Berdasarkan permasalahan tersebut maka dibutuhkan sistem Keyboard Virtual yang memiliki kecerdasan yang di rancang menggunakan Metode Single Shot Detector (SSD). Tujuan Virtual keyboard ini yaitu menggunakan library opencv berbasis Artificial intelligence, Menerapkan landmark pada jari telunjuk dan jari tengah sebagai keypoint klik terhadap Virtual Keyboard dan melakukan pengujian hasil berdasarkan pengaruh jarak dan cahaya terhadap jari tangan. Hasil yang diperoleh dari penelitian ini yaitu masing-masing tombol yang dibuat pada virtual keyboard telah berfungsi dengan baik dan memiliki titik koordinat pada masing-masing tombol virtual keyboard. Pengujian yang telah dilakukan berdasarkan jarak antara 10 cm – 150 cm memiliki akurasi sebesar 86,66%. Sedangkan pengujian sistem berdasarkan intensitas cahaya yang terdiri dari 0,8 lux, 6,0 lux, 26,2 lux, 60 lux, 200 lux dan 627 lux memiliki akurasi sebesar 83,3%. Pengujian kata dilakukan sebanyak 10 kali dan menghasilkan akurasi sebesar 100%. Pengujian tangan kanan dan kiri memiliki akurasi sebesar 100%.

**Kata kunci**— Virtual Keyboard, Artificial Intelligence, Single Shot Detector, landmark

**Abstract**— computer itself certainly requires companion devices, one of which is a keyboard. The keyboard is a computer input device that has an arrangement of letters, numbers, and other control functions, the use of the keyboard is very important because some tasks appear in the form of text. But on the keyboard, some problems often occur, for example, keyboard keys that don't work, besides that the keyboard can type itself without being ordered which is caused by keyboard keys that have sunk. So that the use of the keyboard becomes less efficient. Based on these problems, we need an Artificial Intelligence-Based Virtual Keyboard System Using the Single Shot Detector (SSD) Method. The purpose of this virtual keyboard is Creating a virtual keyboard system using a webcam camera as an alternative to conventional keyboards, To develop interaction between humans and computers in operating the keyboard using a webcam camera and Obtain accuracy values based on word testing, testing using the right and left hands, testing based on the effect of distance and testing based on the influence finger light. The results obtained from this study are that each button made on the virtual keyboard functioned properly and has a coordinate finger point for each virtual keyboard button. The tests that have been carried out based on the distance between 10 cm – 150 cm have an accuracy of 86.66%. While system testing based on light intensity consisting of 00.8 lux, 06.0 lux, 26.2 lux, 60 lux, 200 lux, and 627 lux has an accuracy of 83.3%. The word test was carried out 10 times and resulted in an accuracy of 100%. Right, and left-hand testing has an accuracy of 100%.

**Keywords:** Virtual Keyboard, Artificial Intelligence, Single Shot Detector, landmark

## I. PENDAHULUAN

Pesatnya perkembangan teknologi diiringi dengan perubahan perilaku Masyarakat cenderung memilih untuk melakukan aktivitas-aktivitas secara cepat dan praktis. Kemajuan teknologi tak bisa dihindarkan. Suatu teknologi lahir dengan tujuan untuk memberikan pengaruh yang positif bagi manusia. Sampai saat ini, masyarakat telah menikmati berbagai pengaruh positif dari teknologi, dengan demikian teknologi merupakan suatu penemuan benda dan alat dalam mewujudkan dari realisasi ilmu dan pengetahuan yang dimiliki manusia. Sekarang ini, komputer masih umum digunakan untuk berinteraksi dengan manusia, dimana cara menggunakannya juga sangat mudah hanya dengan menekan tombol pada keyboard atau menyentuh dengan mouse untuk memasukkan dan menerima informasi yang diinginkan. Interaksi manusia dan komputer bertujuan untuk memudahkan pengguna dalam menggunakan komputer, prinsip

pengoperasian komputer atau teknologi adalah adanya input, proses dan output.

Teknologi berbasis gestur merevolusi dunia dan gaya hidup, dan para pengguna merasa nyaman dan peduli akan kebutuhannya, misalnya dalam komunikasi [1]. Perangkat input diperlukan untuk pengoperasian komputer. Piranti ini dapat memasukkan, menyimpan, dan memberikan perintah ke sistem komputer. Komputer pribadi (PC) biasanya memiliki keyboard dan mouse [2]. Interface berkembang dari *command line* ke grafis yang menggunakan keyboard dan mouse sebagai media input. Keyboard merupakan salah satu piranti masukan komputer yang terdiri dari huruf, angka, dan fungsi kontrol lainnya Namun, keyboard sangat penting karena beberapa tugas dilakukan dalam bentuk teks dan dimasukkan ke sistem komputer. Namun pada keyboard terdapat permasalahan yang sering terjadi misalnya tombol keyboard yang tidak berfungsi, selain itu juga keyboard dapat mengetik sendiri tanpa diperintah yang diakibatkan oleh tombol keyboard yang sudah tenggelam. Sehingga penggunaan keyboard menjadi kurang

efisien. Berdasarkan permasalahan tersebut maka dibutuhkan sistem pada komputer agar penggunaannya lebih mudah dan efisien yaitu dengan memanfaatkan kamera *device* atau *webcam* sebagai perangkat masukan secara *realtime* dalam hal ini membuat *virtual keyboard* berdasarkan *hand landmark model*. *Virtual* sendiri menghadirkan pengalaman interaksi menarik yang dapat mengangkut dan memperluas kemampuan pengguna [3]. Jadi, proses input akan efektif jika interaksi dapat dilakukan dengan baik. Pilihan piranti harus mempertimbangkan faktor psikologis seperti kemudahan dan kebiasaan sehari-hari dengan interaksi alami dengan teknologi [4]. Tujuan dari penelitian ini antara lain *Virtual keyboard* menggunakan *library opencv* berbasis *Artificial intelligence*, Menerapkan *landmark* pada jari telunjuk dan jari tengah sebagai keypoint klik terhadap *Virtual Keyboard*, dan melakukan pengujian hasil dilakukan berdasarkan pengaruh jarak dan cahaya terhadap jari tangan.

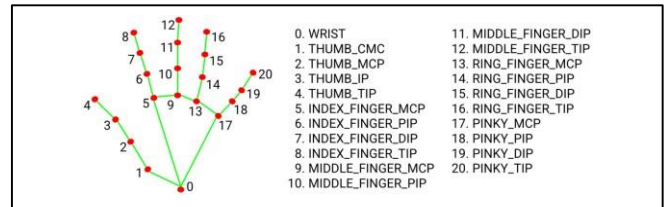
Hasil penelitian, *Virtual Keyboard*, memungkinkan untuk menjadi salah satu opsi di tengah peredaran keyboard konvensional. Sebenarnya, keyboard ini dimulai dengan teknologi optik yang dipatenkan oleh insinyur yang berasal dari IBM (International Business Machines) pada tahun 1992. Keyboard virtual, juga disebut sebagai keyboard laser, sangat bermanfaat bagi mereka yang tidak mampu menggunakan keyboard standar karena berfungsi sebagai pengganti keyboard konvensional. Namun, sebenarnya keyboard virtual melakukan banyak hal.

Python adalah bahasa pemrograman interpretatif multiguna dengan filosofi perancangan yang berfokus pada tingkat keterbacaan kode. Python dianggap sebagai bahasa yang memiliki banyak kemampuan dan kemampuan dengan sintaksis kode yang sangat jelas dan memiliki banyak fitur pustaka standar [6]. Webcam, juga disebut kamera web, adalah sebutan untuk kamera waktu nyata yang gambarnya dapat dilihat melalui internet, program pengolah pesan cepat, atau aplikasi pemanggilan video. Kamera web juga dapat didefinisikan sebagai kamera video digital kecil yang disambungkan ke komputer melalui port USB, port COM, atau jaringan Ethernet atau Wi-Fi [7]. *Flowchart* berfungsi sebagai representasi visual dari suatu algoritma atau prosedur untuk memecahkan suatu masalah; ini memungkinkan pengguna untuk memeriksa aspek analisis masalah yang terlupakan. *Flowchart* memudahkan komunikasi proses program kepada non-programmer [8].

Lebih dari lima ratus fungsi untuk menangani visi komputer tersedia dalam perpustakaan OpenCV, yang berbasis open source. Perangkat lunak ini tersedia di bawah lisensi BSD dan dapat digunakan oleh perusahaan dan organisasi [9]. *MediaPipe* adalah framework C++ yang dapat digunakan untuk membuat aplikasi yang berjalan di berbagai platform, seperti Windows, Linux, MacOS, Android, dan bahkan website. *MediaPipe* dapat memproses banyak data dengan sangat cepat berkat multithreading dan peningkatan kecepatan GPU. Kalkulator *MediaPipe* berjalan saat program dimulai dan berhenti saat program selesai [10]. Salah satu komponen penting dari interaksi manusia-komputer (HCI) adalah pengenalan isyarat tangan (HGR). Model HGR telah banyak digunakan dalam berbagai aplikasi, seperti prostesis cerdas, pengenalan bahasa isyarat, rehabilitasi perangkat, dan kontrol perangkat. [11]

Proses mengurangi penggunaan augmentasi data, seperti rotasi, flip, dan skala, yang dilakukan dalam model Deep Learning,

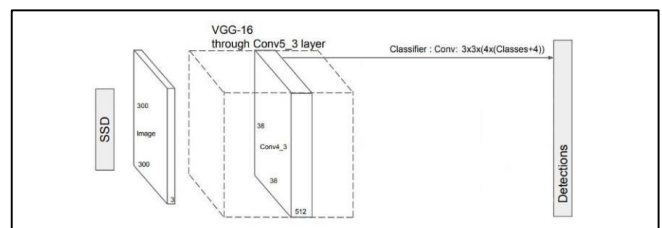
dikenal sebagai *Hand Landmark Model* dan digunakan sebagian besar waktunya. kekuatan lokasi landmark. Cara tradisional adalah dengan mengidentifikasi tangan di dalam bingkai dan kemudian meletakkan landmark di atas bingkai saat ini. Namun, strategi yang berbeda digunakan untuk menangani tantangan pipeline ML di palm detector ini. Mendeteksi tangan membutuhkan banyak waktu karena membutuhkan gambar pemrosesan, ambang batas, dan berbagai ukuran tangan.



Gambar 1 Landmark

*Pyautogui* merupakan bagian dari library Python yang kompatibel dengan berbagai sistem operasi seperti Windows, Mac, dan Linux. Beberapa fitur utama yang tersedia di *PyAutoGUI* adalah driver cursor mouse, klik otomatis atau input pada formulir, tugas keyboard seperti print-screen, menampilkan pesan ke pengguna, dan otomatisasi [12].

Salah satu metode untuk mengenali atau mendeteksi objek pada suatu gambar dengan menggunakan single deep neural network adalah single shot detection (SSD). Ini adalah salah satu algoritma deteksi objek yang paling populer karena kemudahan penggunaan dan akurasi yang tinggi terhadap komputasi yang dibutuhkan [13]. Single Shot Detector (SSD) memungkinkan deteksi objek secara real-time dan mempercepat proses karena tidak memerlukan jaringan proposal wilayah. SSD meningkatkan fitur multi-skala dan kotak default untuk memperbaiki penurunan akurasi. Peningkatan ini memungkinkan SSD untuk mencocokkan akurasi Faster R-CNN menggunakan gambar beresolusi lebih rendah, yang selanjutnya mendorong kecepatan lebih tinggi. SSD terbagi menjadi dua bagian yaitu extract peta fitur terapan fitur konvolusi untuk mendeteksi objek. SSD menggunakan VGG16 untuk mengekstrak peta fitur. Kemudian mendeteksi objek menggunakan lapisan konvolusi. Setiap prediksi terdiri dari kotak batas dan 21 skor untuk setiap kelas (satu kelas tambahan untuk tidak ada objek), dan kami memilih skor tertinggi sebagai kelas untuk objek yang dibatasi. *Conv4\_3* membuat total  $38 \times 38 \times 4$  prediksi: empat prediksi per sel terlepas dari kedalaman peta fitur. Seperti yang diharapkan, banyak prediksi tidak mengandung objek. SSD mencadangkan kelas "0" untuk menunjukkan bahwa ia tidak memiliki objek [14].

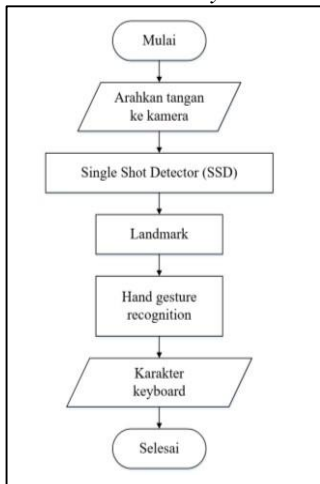


Gambar 2 Single Shot Detector

Gambar citrate adalah gambar dwimarta. Dari perspektif sistematis, citra adalah fungsi kontinu dari intensitas cahaya pada bidang dwimarta (2D). Gambar diam adalah gambar tunggal yang tidak bergerak, sedangkan gambar bergerak adalah rangkaian gambar diam yang ditampilkan secara sekuensial. Namun, gambar digital adalah gambar yang tersusun dalam bentuk raster. Setiap kotak (tile) disebut pixel (picture element) dan memiliki koordinat (x,y). Sumbu x adalah kolom (column), contoh, dan sumbu y adalah baris (row line). Setiap pixel memiliki angka yang menunjukkan intensitas keabuannya. Tingkat keabuan yang menunjukkan level gelap atau kode warna Gambar, atau citra, adalah gambar pada bidang dua dimensi secara harafiah. Namun, dari perspektif matematis, citra adalah fungsi terus-menerus (terusan) dari intensitas cahaya pada bidang dua dimensi [15].

## II. METODOLOGI PENELITIAN

### 1. Flowchart Sistem Virtual Keyboard



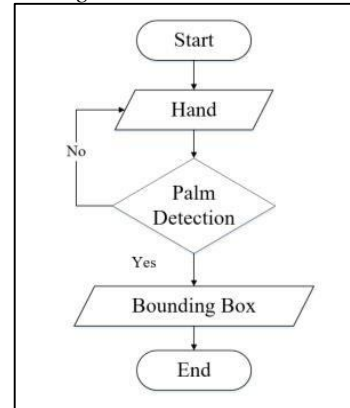
Gambar 3 Flowchart Sistem Virtual Keyboard

Adapun penjelasan dari tahapan *flowchart virtual keyboard* ialah sebagai berikut :

- Arahkan tangan ke kamera**  
Merupakan proses awal yang dilakukan dalam penelitian dengan cara input tangan didepan kamera internal atau webcam secara *realtime*.
- Single Shot Detector**
- SSD dirancang untuk mampu mendeteksi objek secara real-time. R- CNN yang lebih cepat menggunakan jaringan proposal wilayah untuk membuat kotak batas dan menggunakan kotak tersebut untuk mendeteksi objek.
- Landmark**  
Setelah input tangan didepan kamera internal atau *webcam*, kemudian dilakukan pengujian terhadap landmark yang diinputkan kedalam gesture telapak tangan.
- Hand gesture recognition**  
Kemudian proses yang dilakukan adalah dengan melakukan gerakan tangan terhadap objek yang akan dituju dengan landmark yang menjadi kontrol *keyboard*.

- Hasil**  
Hasil merupakan *output* berupa karakter yang dihasilkan dari *virtual keyboard* berdasarkan gerakan jari tangan berbasis landmark sebagai kontrol interaksi dengan komputer.

### 2. Flowchart Single Shot Detector

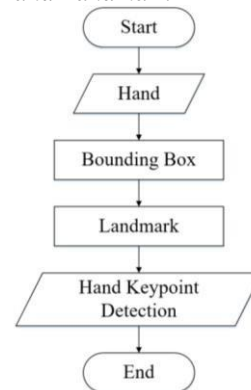


Gambar 4 Flowchart Single Shot Detector

Adapun penjelasan dari tahapan *flowchart single shot detector (SSD)* ialah sebagai berikut:

- Hand**  
Inputan merupakan sebuah objek berupa telapak tangan yang akan di deteksi oleh kamera kemudian akan masuk kedalam kondisi berupa palm detection.
- Palm Detection**  
Berikut ini adalah kondisi dari palm detection yaitu pendeteksian telapak tangan, apabila objek tangan tidak terdeteksi maka proses diulang dan apabila objek tangan terdeteksi maka *output* yang dihasilkan berupa *object* tangan yang terdeteksi dan dibungkus kedalam kotak pembatas atau *bounding box*.
- Bounding Box**  
Hasil *output* dari proses *single shot detector (SSD)* yaitu *bounding box* yang merupakan kotak pembatas yang menandakan tangan telah terdeteksi.

### 3. Flowchart Hand Landmark



Gambar 5 Flowchart Hand Landmark

Adapun penjelasan dari tahapan *flowchart Hand Landmark* ialah sebagai berikut:

- a. *Hand*  
*Flowchart hand landmark* model merupakan alur dari proses *landmark* dimana terdiri dari objek berupa telapak tangan sebagai inputan.
- b. *Bounding Box*  
*Bounding box* yang merupakan telapak tangan yang telah terdeteksi dan disertai kotak pembatas persegi sebagai tangan yang telah terdeteksi.
- c. *Landmark*  
Setelah deteksi telapak tangan terjadi kemudian menuju pada *landmark* yang merupakan model tenggara tangan.
- d. *Hand Keypoint Detection*  
Hasil *outputnya* berupa 21 titik koordinat setiap titik jari berbentuk 3D di dalam wilayah telapak tangan yang terdeteksi.



Gambar 6 Proses single shot detector dan landmark

### 2. Proses Perhitungan Klik Tombol Menggunakan Model Landmark

Proses perhitungan dilakukan dengan menggunakan model landmark. Tahap awal dalam perhitungan ini ialah mendapatkan koordinat dari jari yang digunakan untuk proses klik terhadap tombol, kemudian mendapatkan nilai dari proses klik terhadap tombol. Adapun rumus yang digunakan ialah seperti berikut :

$$n = [ x_2 - x_1 ] [ y_2 - y_1 ]$$

Keterangan :

$n$  = objek (huruf, angka, karakter pada keyboard)

$x_1$  = nilai koordinat  $x_1$  (keypoint 7)

$x_2$  = nilai koordinat  $x_2$  (keypoint 8)

$y_1$  = nilai koordinat  $y_1$  (keypoint 7)

$y_2$  = nilai koordinat  $y_2$  (keypoint 8)

Kemudian menghitung nilai dari tombol yang ditekan menggunakan model landmark yang sudah didapat menggunakan rumus dibawah ini :

$$xy^2 = x^2 + y^2$$

Keterangan :

$xy$  = nilai objek yang akan dicari (angka, huruf, karakter)

$x$  = nilai koordinat  $x$  pada objek

$y$  = nilai koordinat  $y$  pada objek

### 3. Tampilan Sistem Virtual Keyboard

Tampilan awal sistem *virtual keyboard* berupa rangkaian tombol-tombol *keyboard* seperti halnya pada *keyboard* fisik yang telah didesain tampilannya menggunakan *library opencv* dan ditampilkan melalui kamera *internal* maupun kamera *webcam* laptop dan menggunakan notepad sebagai lembar kerja pengetikan dari sistem *virtual keyboard* tersebut.



Gambar 7 Tampilan Sistem

Setelah sistem *virtual keyboard* tampil pada layar *desktop* laptop, maka peneliti akan melakukan kontrol *keyboard virtual* menggunakan telapak tangan dengan cara

## III. HASIL DAN PEMBAHASAN

### 1. Single Shot Detector dan Landmark Model

*Landmark* merupakan proses pemberian titik tenggara pada telapak tangan yang terdeteksi oleh kamera *internal* maupun kamera *webcam* pada laptop. Proses pendeteksian telapak tangan pada penelitian ini menggunakan metode *Single Shot Detector (SSD)*. Tahapan *Single Shot Detector (SSD)* akan menghasilkan *output* berupa *bounding box* (kotak pembatas) pada telapak tangan. Proses ini dimulai dengan inputan awal berupa objek yakni telapak tangan yang diarahkan ke depan kamera *internal* maupun *webcam* laptop, kemudian menuju ke tahap konvolusi yang merupakan lapisan ekstraksi fitur pada objek yang terdeteksi. Semakin menurunnya lapisan *convolutional* maka akan mengurangi ukuran peta fitur dan meningkatkan kedalaman. Lapisan konvolusi awal mencakup bidang reseptif yang lebih kecil dan sangat membantu dalam mendeteksi objek yang lebih kecil yang ada dalam gambar, proses selanjutnya ialah mendeteksi telapak tangan untuk memperkirakan kotak pembatas berbentuk persegi, proses ini disebut *palm detector*. Hasil akhir dari proses *single shot detector (SSD)* ialah *bounding box* atau kotak pembatas yang menandakan bahwa objek telah terdeteksi.

Selanjutnya proses dilanjutkan dengan pemberian *landmark* pada telapak tangan untuk proses pengetikan pada *virtual keyboard*. langkah awal dimulai dengan mengarahkan telapak tangan pada kamera *internal* atau kamera *webcam*, kemudian proses *bounding box* yakni telapak tangan disertai kotak pembatas yang telah terdeteksi oleh kamera dan terakhir pemberian *landmark*, apabila *landmark* terdeteksi maka proses dilanjutkan ke proses akhir. Hasil akhir proses ini berupa *keypoint* pada setiap titik dari telapak tangan hingga seluruh jari tangan, total titik yang terdeteksi sebanyak 21 titik. *Landmark* inilah yang nantinya digunakan untuk mengontrol *virtual keyboard*. gambar proses *single shot detector* disertai *landmark* dapat dilihat seperti gambar 4.1(a) Hand, 4.1(b) Hand Detection, dan 4.1(c) Landmark.

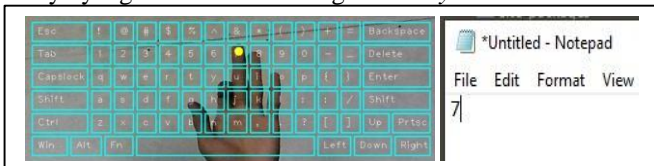


mengarahkan telapak tangan ke depan kamera *internal* atau kamera *webcam*, setelahnya maka telapak tangan akan terdeteksi di dalam sistem *virtual keyboard* dengan jari tengah yaitu pada *keypoint* 12 disertai dengan *shape* lingkaran sebagai pendeteksi *keyboard* yang akan di klik. Jika tangan sudah terdeteksi barulah peneliti dapat melakukan kontrol *keyboard virtual* tersebut dimana hasil pengetikan dari angka, huruf dan karakter akan tampil di lembar kerja pada notepad maupun lembar kerja yang membutuhkan sistem pengetikan *keyboard*.



Gambar 8 Tampilan tangan yang terdeteksi didalam sistem virtual keyboard

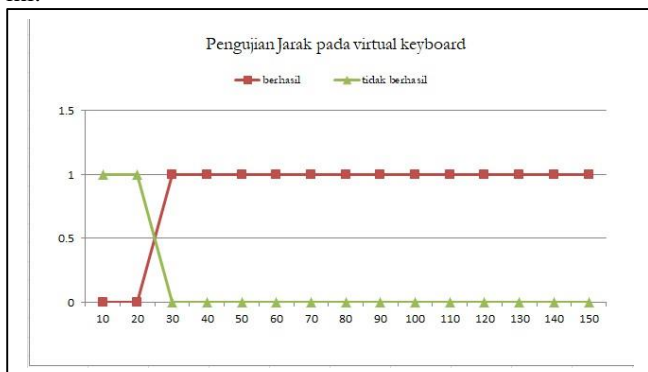
Selanjutnya setelah tangan terdeteksi didalam sistem virtual keyboard maka langkah selanjutnya melakukan kontrol keyboard, pada sistem ini pengetikan dilakukan dengan menggunakan 2 jari yaitu jari telunjuk dan jari tengah, jari tengah disertai *shape* lingkaran berwarna merah yang berfungsi untuk mendeteksi huruf ataupun tombol yang ada pada *virtual keyboard*, jari telunjuk akan digunakan untuk fungsi klik pada *keyboard*, ketika fungsi *keyboard* berjalan maka *shape* yang ada pada jari tengah akan berubah warna menjadi berwarna kuning. Hasil pengetikan dari angka, huruf dan karakter akan tampil di lembar kerja notepad, selain notepad *output* juga dapat dihasilkan menggunakan *software* lainnya seperti *microsoft office*, *search engine* dan *software* lainnya yang membutuhkan fungsi dari *keyboard* sendiri.



Gambar 9 Tampilan pengetikan virtual keyboard

#### 4. Pengujian Jarak Pada Virtual Keyboard

Pengujian jarak tangan terhadap kamera pada sistem virtual keyboard dilakukan berdasarkan jarak 10 cm – 110 cm yaitu terdiri dari 10 cm, 20 cm, 30 cm, 40 cm, 50 cm, 60 cm, 70 cm, 80 cm, 90 cm, 100 cm dan 110 cm yang dilakukan diluar ruangan dapat dilihat pada grafik dibawah ini.



Gambar 10 Grafik pengujian jarak

Berdasarkan grafik diatas hasil pengujian jarak terhadap kamera pada sistem virtual keyboard diperoleh 13 status berhasil pada jarak 30 cm – 150 cm dimana sistem dapat berjalan dengan telapak tangan terdeteksi dan dapat mengontrol keyboard secara virtual dan diperoleh 2 status tidak berhasil dengan jarak 10 cm dan 20 cm, dengan jarak 10 cm dan 20 cm telapak tangan masih belum terdeteksi karena tangan terlalu dekat dengan kamera sehingga tangan tidak sepenuhnya tampak pada sistem virtual keyboard. Jadi tingkat akurasi keberhasilan pengujian jarak tangan terhadap kamera pada sistem virtual keyboard adalah:

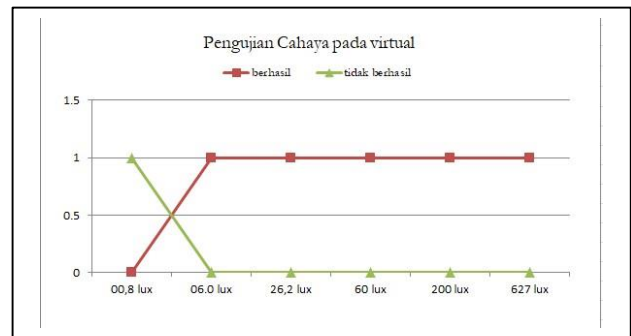
$$\text{Akurasi} = \frac{(\text{Total pengujian berhasil})}{(\text{Total data pengujian})} \times 100\%$$

$$\text{Akurasi} = 13/15 \times 100\% = 86,66\%$$

Tingkat akurasi keberhasilan pengujian jarak terhadap kamera pada sistem virtual keyboard diperoleh akurasi sebesar 86.66%.

#### 1. Pengujian Cahaya pada Virtual Keyboard

Pengujian berdasarkan pengaruh intensitas cahaya terhadap sistem virtual keyboard terdiri dari 00,8 lux, 06.0 lux, 26,2 lux, 60 lux, 200 lux dan 627 lux, dapat dilihat pada grafik dibawah ini.



Gambar 11 Grafik pengujian cahaya

Berdasarkan grafik diatas diperoleh 1 status tidak berhasil pada intensitas cahaya 00,8 yang dilakukan pada malam hari didalam ruangan dan lampu mati dan minim pencahayaan, status tidak berhasil karena intensitas cahaya yang terlalu gelap sehingga telapak tangan tidak dapat terdeteksi, sedangkan untuk intensitas cahaya yang lainnya telapak tangan dapat terdeteksi sehingga dapat mengontrol keyboard.

Jadi tingkat akurasi keberhasilan terhadap pengujian intensitas cahaya pada sistem virtual keyboard adalah:

$$\text{Akurasi} = \frac{5}{6} \times 100\% = 83,3\%$$

Tingkat akurasi keberhasilan terhadap pengujian berdasarkan intensitas cahaya menggunakan kamera webcam dan lux meter yang terdiri dari 00,8 lux, 06.0 lux, 26,2 lux, 60 lux, 200 lux dan 627 lux. adalah 83,3%.

#### IV. KESIMPULAN

Kesimpulan berisi tentang poin-poin utama artikel.

Dari hasil penelitian dengan judul Implementasi Metode Single Shot Detector (SSD) Untuk Pengenalan Gerakan Tangan Sebagai Input Virtual Keyboard dapat disimpulkan sebagai berikut :

1. Masing-masing tombol yang dibuat pada virtual keyboard telah berfungsi dengan baik dan memiliki titik koordinat pada masing-masing tombol virtual keyboard.
2. Pengujian yang telah dilakukan berdasarkan jarak antara 10 cm – 150 cm memiliki akurasi sebesar 86,66%.
3. Pengujian sistem berdasarkan intensitas cahaya menggunakan kamera internal yang terdiri dari 0,8 lux, 6,0 lux, 26,2 lux, 60 lux, 200 lux dan 627 lux memiliki akurasi sebesar 83,3%.
4. Pengujian kata dilakukan sebanyak 10 kali dan menghasilkan akurasi sebesar 100%.
5. Pengujian tangan kanan dan kiri memiliki akurasi sebesar 100%.

#### REFERENSI

- [1] Rahim. M. A and Shin. J. (2020). Hand Movement Activity-Based Character Input System on a Virtual Keyboard. The University of Aizu..
- [2] 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.
- [3] Dudley, J. J, Benko, H., Wigdor, D., & Krisstenson, U. O. (2019). Performance Envelopes of Virtual Keyboard Text Input Strategies in Virtual Reality. *IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*
- [4] Falabiba. (2019). *Modul Interaksi Manusia dan Komputer*.
- [5] Efendi, 2020. “Mengenal keyboard Virtual” *NESABAMEDIA*.
- [6] Melinda. R. N., et al. 2021. “Program Perhitungan RAB Pekerjaan Struktur Baja (WF BEAM) Menggunakan Bahasa Python”. *TIERS Information Technology Journal*. 2(1): 1-7.
- [7] Hermansyah. M dan Fadillah. N., (2019). “Virtual Mouse Berdasarkan Warna RGB Menggunakan Metode Optical Flow Secara Real-Time”. *Paradigma – Jurnal Informatika dan Komputer*. 21(2). 249-252.
- [8] Solikin. (2020). Deteksi Penyakit Pada Tanaman Mangga Dengan Citra Digital : Tinjauan Literatur Sistematis ( SLR ). *Bina Insani ICT Journal*, 7(1), 63–72.
- [9] Kadir, Abdul. 2019. Langkah Mudah Pemrograman OpenCV & Python. Jakarta: PT. Elex Media Komputindo.
- [10] F. Zhang et al., "MediaPipe Hands: On-Device Real-Time Hand Tracking," *ArXiv*, vol. abs/2006.10214, 2020.
- [11] Asmoro. J. D.(2022). *Virtual Mouse Menggunakan Hand Gesture Recognition Berbasis Hand Landmark Model Untuk Pointing Device*.
- [12] Pradono et al., Development of automatic Landsat data download engine. *Materials Science and Engineering*.
- [13] Fuady Samratul dan Anggraeni Gina. (2020). Deteksi Objek Menggunakan Metode *Single Shot Multibox Detector* Pada Alat Bantu Tongkat Tunanetra Berbasis Kamera. *Journal of Electrical Power Control and Automationn*, 3(2).
- [14] Hui Jonatahan. (2018). SSD object detection: Single Shot MultiBox Detector for real-time processing.
- [15] Raharja. B. D., et all. 2021. “Penerapan Discrete Cosine Transform (Dct) Terhadap Kompresi Citra Digital”. *Indonesian Journal of Business Intelligence*. 4(1). 31-36