

SISTEM DETEKSI SOCIAL DISTANCING MENGGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK (CNN)

Samuel Hadi Winata²

Teknik Informatika, Universitas 17 Agustus 1945 Surabaya

Email Correspondence Author : samuelwinata32@gmail.com

Abstract

COVID-19 is a virus that is currently being faced by more than 180 countries around the world. Many efforts have been made to narrow the spread of the COVID-19 virus, one of which is social distancing. Social distancing is a recommendation that everyone limit themselves from being in crowded places, and to limit at least a distance of two meters from everyone around them. However, there are still many parties who are not responsible for the violation of this social distancing health protocol, it is not uncommon to find people who are huddled together, both in work areas, entertainment venues, and entertainment/recreational places. Therefore, in this final project, a social distance detection system using the Convolutional Neural Network (CNN) method using the YOLO architecture will be developed which can detect objects with an accuracy of 91% with 2000 iterations of training. So that this system can be used to see people in real time who spread social distance and people who are less than two meters in order to increase discipline to fight the COVID-19 virus.

Keyword: *Social Distancing Detection, Convolutional Neural Network (CNN) Method, YOLO.*

Abstrak

COVID-19 merupakan virus yang sedang dihadapi saat ini oleh lebih dari 180 negara di seluruh dunia. Banyak upaya dilakukan untuk mempersempit penyebaran virus COVID-19, salah satunya dengan *social distancing*. *Social distancing* merupakan sebuah anjuran agar setiap orang membatasi diri untuk tidak berada di tempat yang ramai / berkerumun, serta membatasi paling tidak jarak dua meter dari setiap orang disekitarnya. Namun masih saja banyak pihak yang tidak bertanggung jawab melanggar protokol kesehatan *social distancing* ini, tidak jarang ditemukan kerumunan masyarakat yang berhimpitan, baik di area pekerjaan, tempat perbelanjaan, dan tempat hiburan/rekreasi. Oleh karena itu pada tugas akhir ini akan dikembangkan sebuah sistem pendeteksi social distancing dengan metode *Convolutional Neural Network* (CNN) menggunakan arsitektur YOLO yang dapat mendeteksi objek dengan akurasi ketepatan sebesar 91% dengan training sebanyak 2000 iterasi. Sehingga sistem ini dapat digunakan untuk melihat secara real time orang-orang yang menaati anjuran *social distancing* dan menandai orang-orang yang berjarak kurang dari dua meter agar dapat ditingkatkan kedisiplinan guna memberantas virus COVID-19.

Kata Kunci: *Deteksi Social Distancing, Metode Convolutional Neural Network (CNN), YOLO.*

I. PENDAHULUAN

Kemajuan teknologi semakin pesat di era globalisasi ini membawa perubahan yang sangat besar di berbagai sektor kehidupan manusia. Namun tidak menutup kemungkinan bahwa dengan berkembangnya seluruh teknologi yang ada, juga berdampak dengan permasalahan yang timbul. Beberapa bulan kebelakang tepatnya pada 17 November 2019 diduga merupakan kasus pertama terjadinya pandemi virus di China dan berangsur-angsur menyebar ke seluruh penjuru dunia yang menyebabkan kekacauan dalam berbagai sektor kehidupan manusia. Virus yang menyerang sistem pernafasan manusia dan bisa berakibat kematian ini disebabkan oleh virus Corona COVID-19.

COVID-19 merupakan pandemi yang baru saja terjadi di lebih dari 180 negara dengan total korban meninggal mencapai 1,48 juta jiwa. Dampak dari pandemi ini dapat dirasakan oleh berbagai kalangan tanpa terkecuali. Dikarenakan vaksin dari virus COVID-19 ini belum direalisasikan maka langkah yang diambil untuk mengurangi penyebaran virus COVID-19 ini antara lain menerapkan protokol kesehatan mulai dari mencuci tangan sebelum/sesudah melakukan aktivitas, memakai masker saat berpergian, dan menerapkan *social distancing* dari orang-orang disekitar. Penerapan *social distancing* sendiri dapat menurunkan angka penularan sekitar 3%. Namun masih banyak masyarakat yang menganggap *social distancing* tidak begitu penting karena mereka sudah memakai masker dan mencuci tangan sebelum/sesudah beraktivitas.

Oleh karena itu dengan didukung oleh kemajuan teknologi yang makin pesat dan beberapa penelitian terkait yang sudah dihasilkan maka akan dilakukan pengimplementasian sebuah sistem deteksi *social distancing* dengan metode *Convolution Neural Network* (CNN) menggunakan

arsitektur YOLO. Diharapkan dengan adanya sistem tersebut dapat memantau kerumunan orang yang ada disuatu tempat secara real time untuk mengetahui masyarakat yang tetap menjalankan protokol *social distancing*.

II. DASAR TEORI

1. *Social Distancing*

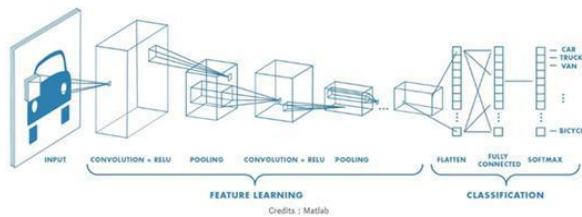
Social distancing adalah sebuah bentuk tindakan untuk mengupayakan penurunan penularan penyakit dengan cara memperlebar jarak antar tiap orang yang satu dengan orang lainnya. Pembatasan Sosial Berskala Besar (PSBB) telah dilakukan oleh negara Indonesia sebagai bentuk upaya yang dilakukan untuk pendisiplinan *social distancing* bagi masyarakatnya. Pembatasan Sosial Berskala Besar (PSBB) adalah pembatasan yang terkait dengan kegiatan maupun pekerjaan tertentu seluruh penduduk dalam suatu area tertentu yang diduga terdampak virus COVID-19 untuk pencegahan yang memungkinkan terjadinya penyebaran virus COVID-19[1].

Menjaga jarak antar social merupakan salah satu tindakan yang dianggap dapat mengurangi resiko penularan virus corona atau COVID-19 pada masa ini. *Social distancing* dinilai dapat mengurangi resiko terkait dengan semakin meluasnya penyebaran virus COVID-19 dikarenakan virus ini dapat menular melalui interaksi antar manusia disaat orang yang positif menderita flu berpapasan dengan orang lain yang sehat. Penerapan *social distancing* yang biasanya dilakukan oleh masyarakat yaitu bekerja dari rumah masing-masing, penerapan belajar online di rumah masing-masing, dan juga menunda untuk bepergian ke tempat ramai yang dinilai banyak terjadinya kerumunan orang[2].

2. *Convolutional Neural Network* (CNN)

Convolutional Neural Network adalah salah satu dari berbagai jenis *neural network* yang sering dipergunakan dalam proses pengolahan citra atau data image. Konvolusi

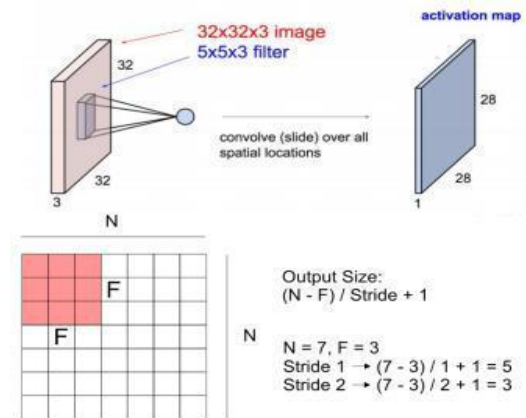
atau kebanyakan sering disebut sebagai *convolution* adalah sebuah matriks yang berfungsi untuk melakukan filterisasi. *Convolutional Neural Network* mempunyai banyak *layer-layer* dimana *layer* ini dapat difungsikan untuk untuk melakukan filter ke tiap proses yang dikerjakannya. Proses ini sering disebut dengan proses *training*. Pada tahapan proses *training* terdapat tiga jenis tahap yaitu *Convolutional layer*, *Pooling layer*, dan *Fully connected layer* [3].



Gambar 1. Proses Convolutional Neural Network (Santoso & Ariyanto, 2018).

2.1 Convolutional Layer

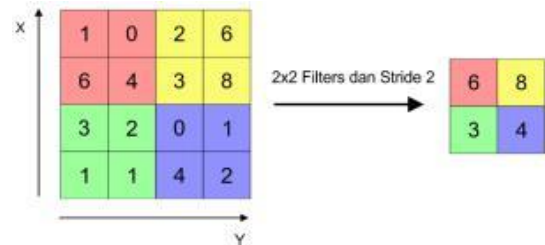
Data menyeluruh yang nantinya akan melewati lapisan konvolusional ini akan mengalami proses konvolusi. Lapisan ini nantinya akan melakukan konversi pada setiap filter yang akan diproses ke seluruh bagian dari data yang masuk dan akan menghasilkan sebuah *feature map* 2D atau *activation map*. Filter-filter yang ada pada Layer ini memiliki panjang, tinggi (piksel) dan ketebalan yang akan sesuai dengan tiap channel data masukan. Setiap filter yang ada nantinya akan mengalami sebuah proses pergerakan dan operasi “dot” diantara data masukan dan nilai yang ada pada filter tersebut. Lapisan konvolusional ini nantinya secara signifikan akan mengalami kompleksitas terhadap model melalui optimalisasi outputnya. Depth, stride dan zero padding merupakan beberapa parameter yang ada untuk melakukan pengoptimalan pada proses tersebut[3].



Gambar 2. Convolutional Layer (Santoso & Ariyanto, 2018).

2.2 Pooling Layer

Pooling Layer adalah tahapan yang akan dilakukan setelah melalui proses tahap sebelumnya yaitu Convolutional Layer. Pooling Layer sendiri adalah sebuah filter yang memiliki ukuran dan juga stride yang sudah ditentukan. Setiap ada pergeseran maka akan ditentukan oleh banyaknya jumlah stride yang akan mengalami pergeseran pada semua area activation map atau feature map.



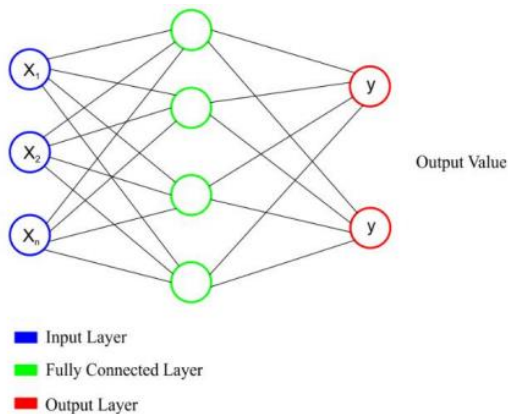
Gambar 3. Pooling Layer (Max Pooling) (Santoso & Ariyanto, 2018).

Didalam penerapan proses Pooling Layer jenis yang sering digunakan adalah jenis Max Pooling (menggunakan pooling dari nilai terbesar) dan Average Pooling (menggunakan pooling dari nilai rata-rata). Sebagai contohnya, apabila yang akan kita gunakan adalah Max Pooling 2x2 dengan nilai Stride 2, dimana di setiap filter yang melakukan pergeseran, nilai yang akan diambil adalah nilai paling besar pada area 2x2, Namun ketika menggunakan

Average Pooling maka yang akan diambil adalah nilai rata-rata pada area 2x2[3].

2.3 Fully Connected Layer

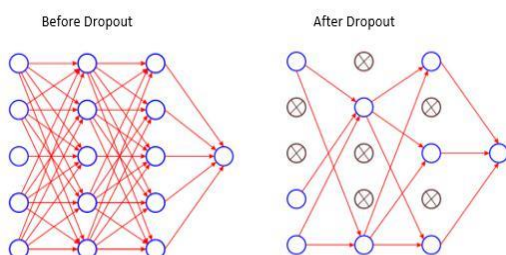
Fully Connected Layer merupakan Feature Map yang berhasil didapatkan dari tahapan sebelumnya dengan memiliki bentuk multidimensional array. Sehingga nantinya, sebelum akan masuk kepada tahapan Fully Connected Layer, Feature Map tersebut akan mengalami proses “flatten” atau biasa disebut reshape terlebih dahulu[3].



Gambar 4. Fully Connected Layer (Santoso & Ariyanto, 2018).

2.4 Dropout

Dropout adalah proses yang dilakukan untuk melakukan pencegahan terjadinya overfitting dan dropout juga dilakukan untuk mempercepat proses saat dilakukan learning. Overfitting sendiri adalah sebuah kondisi yang terjadi dimana seluruh data yang telah melewati proses training dapat memperoleh nilai persentasi yang baik, tetapi saat akan memprediksi akan mengalami ketidaksesuaian. Dalam sebuah sistem kerja pada tahapan dropout akan menghilangkan untuk sementara waktu suatu neuron yang berupa Hidden Layer yang berada dalam sebuah jaringan[3].



Gambar 5. Sebelum dropout (sebelah kiri) dan setelah dropout (sebelah kanan) (Santoso & Ariyanto, 2018).

3. You Only Look Once (YOLO)

YOLO adalah pendekatan terbaru untuk penerapan sistem pendeteksian sebuah objek, dimana akan ditargetkan untuk melakukan proses secara real-time. YOLO bekerja dengan memberikan bingkai pendeteksian objek sebagai masalah regresi tunggal, dimana dari tiap piksel pada sebuah gambar yang diproses akan langsung diberikan kotak pembatas atau bounding box yang terpisah dan probabilitas dari kelas yang berkaitan dengan objek tersebut. YOLO dapat melakukan proses pengenalan dan pendeteksian pada objek dengan adanya sebuah jaringan syaraf tunggal (single neural network), dimana jaringan syaraf tunggal ini yang akan melakukan prediksi terhadap kotak pembatas atau bounding box dan probabilitas kelas untuk tiap objek secara real-time[4]. Dimana untuk memperoleh nilai prediksi final, faktor yang menentukan adalah dengan cara mencari class confidence score, didasari dengan probabilitas kondisional kelas dan box confidence score.

Class confidence score digunakan untuk mengukur sebuah nilai kepercayaan terhadap proses lokalisasi dan klasifikasi objek yang dideteksi. Class confidence score memberikan sebuah nilai kepercayaan kelas secara spesifik untuk tiap kotak pembatas, dimana akan mengkodekan kemungkinan dari kelas yang akan muncul di kotak pembatas dan seberapa besarnya kesesuaian kotak yang telah diprediksi dengan objeknya.

$$Pr(Class_i|Object) * Pr(Object) * IOU_{pred}^{truth} = Pr(Class_i) * IOU_{pred}^{truth}$$

Keterangan:

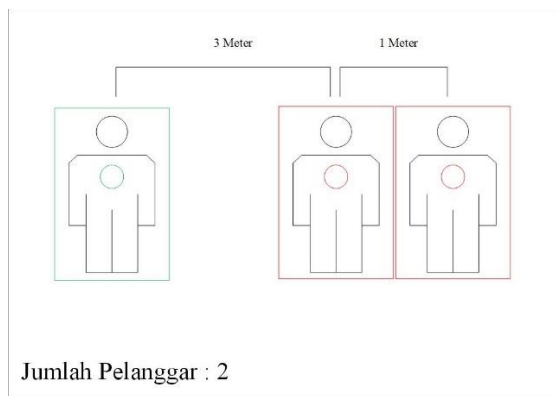
$P_i(Class_i|object)$ adalah probabilitas kondisional kelas i .

$P_i(Class_i)$ adalah probabilitas kelas i .

III. METODOLOGI PENELITIAN

1. Tahapan Persiapan

Tahapan persiapan dimulai dengan mengumpulkan studi literatur dari 5 jurnal yang kurang dari 5 tahun terakhir. Setelah itu penulis mulai membaca, memahami, dan mengumpulkan informasi terkait dengan topik yang akan diteliti. Setelah mendapat seluruh informasi penulis mulai merancang rumusan masalah, batasan masalah, dan tujuan pada penelitian ini. Selanjutnya melakukan perencanaan terkait dengan kebutuhan pada saat penelitian yang berkaitan dengan objek yang diteliti, alat-alat yang digunakan, dan pembuatan desain sistem deteksi *social distancing* menggunakan Convolutional Neural Network (CNN) dengan arsitektur YOLO.



Gambar 5. Desain Sistem Deteksi Social Distancing.

2. Tahapan Implementasi Sistem

Tahapan implementasi sistem dimulai dengan menginputkan dataset berupa 200 gambar yang sebelumnya sudah diambil dari CCTV di PT. Pratama Sumber Milindo Surabaya. Kemudian dataset tersebut dibagi menjadi 2 yaitu 70% sebagai data pelatihan (140 gambar) dan 30% sebagai data uji (60 gambar) yang nantinya di ubah kedalam bentuk bingkai-bingkai sehingga dapat dilakukan pelatihan dan pengujian menggunakan metode Convolutional Neural Network (CNN) dengan arsitektur YOLO.

Setelah itu akan dihasilkan *bounding box* pada setiap objek yang diteliti (objek difokuskan pada manusia) serta menetapkan sentroid pada setiap *bounding box*. Setelah itu dilakukan pengukuran antar 1 objek dengan objek lain yang berpusat pada sentroid masing-masing. Jika jaraknya kurang dari 2 meter atau 6 kaki, maka objek tersebut dikategorikan ke pelanggaran. Setiap *bounding box* nantinya akan diberi warna hijau, namun ketika objek tersebut masuk ke dalam kategori pelanggaran maka *bounding box* akan berubah menjadi warna merah.

3. Tahap Evaluasi

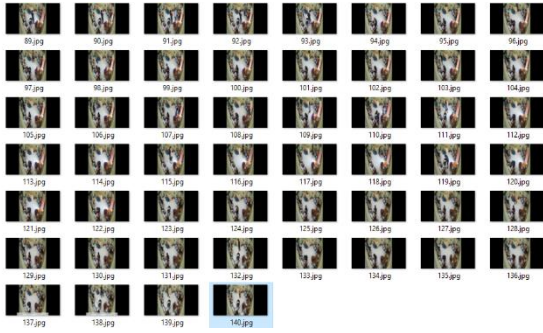
Evaluasi dilakukan dengan cara melakukan implementasi terhadap sistem dengan memasukan 30% data testing atau sebanyak 60 gambar. Sehingga nantinya dapat dihitung keberhasilan suatu sistem dalam melakukan pendeteksian terhadap data testing yang disediakan. Untuk menghitung tingkat keberhasilan maka dapat dihitung dengan persamaan berikut :

$$\text{Tingkat Akurasi Sistem} = \frac{\text{Jumlah data terdeteksi}}{\text{Total jumlah data}}$$

IV. HASIL DAN PEMBAHASAN

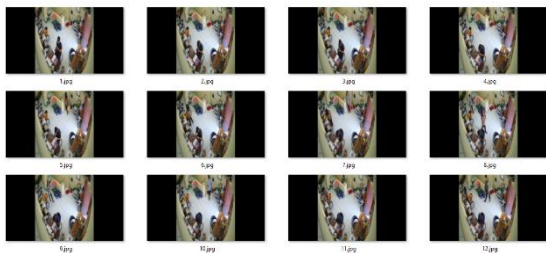
1. Pendistribusian Dataset

Proses awal yang harus dilakukan dalam pembuatan sistem deteksi *social distancing* menggunakan *convolutional neural network* ini adalah mengumpulkan dataset dan membaginya menjadi data *training* dan data *testing*. Dataset yang terdiri dari 200 gambar ini diambil menggunakan CCTV yang berada di PT. Pratama Sumber Milindo Surabaya dengan posisi CCTV yang berada di toko dan ruang administrasi. Sebanyak 70% gambar akan digunakan sebagai data *training* (140 gambar) dan 30% sisanya akan digunakan untuk data *testing* (60 gambar).



Gambar 6. Data training.

Dengan data yang akan dilakukan *training* berjumlah 140 gambar, maka akan dibagi menjadi dua yaitu 70 gambar data *training* diambil dari toko dan 70 gambar sisanya diambil dari ruang administrasi. Kemungkinan lebih banyak objek yang berbeda-beda lebih besar berada di toko karena konsumen yang berganti masuk untuk membeli produk yang ada dengan menampilkan gerakan, warna baju, dan aksesoris yang dikenakan secara berbeda-beda. Sementara di ruang administrasi objek hanya terdiri dari orang-orang yang sama namun dengan gerakan, warna baju, dan aksesoris yang dikenakan secara berbeda juga.



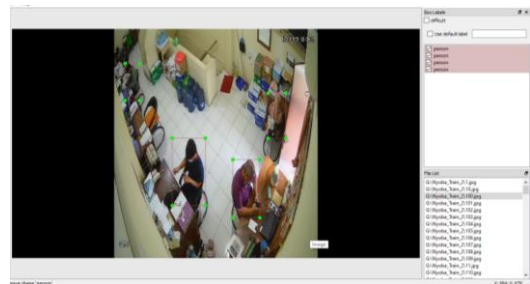
Gambar 7. Data testing

Data *testing* diambil di hari yang berbeda dengan jumlah total sebanyak 60 gambar. Seperti data *training*, data *testing* ini juga diambil dari dua tempat yang berbeda, 30 gambar diambil dari

toko dan 30 gambar diambil dari ruang administrasi.

2. Training dengan YOLO V3

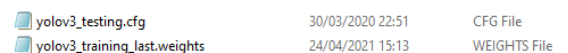
Setelah dataset dibagi sedemikian rupa sehingga mendapat data *training* sebanyak 140 gambar maka akan dilakukan proses *training* terhadap data tersebut. *Training* dilakukan dengan menggunakan configure Darknet network for *training* YOLOv3.



Gambar 8. Memberikan label pada tiap data training.

Tahapan awal yang dilakukan yaitu memberikan label untuk setiap objek. Tiap gambar pada data training untuk seluruh objek manusia akan diberikan label yang diberi nama “person” dan diberikan kotak yang nantinya akan dilakukan proses training pada tahap selanjutnya. Pada Gambar 8 terdapat 4 objek yang akan diteliti sehingga keempatnya dilabeli sebagai objek dengan nama “person”.

Proses training dengan menggunakan *convolutional neural network* menggunakan *framework* YOLOv3 dengan file .cfg. Setelah melakukan proses training dengan iterasi sebanyak 1000 iterasi dan 2000 iterasi, maka akan menghasilkan model yang akan diberi nama *yolov3_training_last.weights* seperti Gambar 9.



Gambar 9. Modul hasil training.

3. Pengimplementasian Sistem

Pada tahap implementasi sistem ini akan dicoba untuk menginputkan data testing sehingga dapat dilihat apakah sistem yang sudah dibuat bisa mendeteksi objek dengan baik. Sistem berjalan dengan mengambil data training untuk mendeteksi objek, selanjutnya sistem akan mulai memberikan bounding box untuk setiap objek yang terdeteksi dengan tambahan sentroid pada bagian tengah objek. Lalu dilakukan pengukuran jarak antara satu sentroid dengan sentroid lain sehingga didapatkan nilai dari euclidian dimana untuk jarak yang kurang dari 2 meter akan digolongkan ke pelanggar.



Gambar 10. Implementasi sistem data testing no. 4.

Seperti pada Gambar 10 dapat dilihat bahwa sistem sudah dapat mendeteksi dengan baik. Namun masih ada beberapa objek (orang) yang tidak terdeteksi. Beberapa objek tidak terdeteksi dikarenakan objek sedang melakukan gerakan seperti jongkok, menghadap belakang, menggunakan baju dengan warna berbeda dan menggunakan aksesoris seperti helm atau kacamata.



Gambar 11. Implementasi sistem data testing eror no. 20.

Pada Gambar 11 terdapat 4 objek (orang) yang masuk ke dalam 1 frame, namun terdapat 2 objek yang tidak terdeteksi dikarenakan ada 1 objek yang melakukan aktivitas jongkok sehingga tidak terdeteksi oleh sistem, dan 1 objek lain tidak terdeteksi dikarenakan jauh dari jangkauan CCTV.

Maka dari keseluruhan data testing yang dilakukan percobaan pendeteksi pada sistem akan dibagi menjadi 2 jenis yaitu percobaan pendeteksian menggunakan data training dengan 1000 iterasi dan 2000 iterasi sehingga dapat diambil hasil sistem pendeteksi social distancing lebih maksimal bila menggunakan 1000 iterasi atau 2000 iterasi.

Data Testing No. 5	
Iterasi 1000	Iterasi 2000
<p>Penjelasan: Terdapat perbedaan dari kedua iterasi pada data testing no. 5, dimana pada iterasi 2000 terdapat objek yang terdeteksi sehingga pada 1 frame terdapat 3 orang yang berhasil terdeteksi oleh sistem. Satu objek pada iterasi 1000 tidak terdeteksi dengan baik dikarenakan objek terletak tersembunyi disebelah pintu. Namun dengan menggunakan iterasi 2000 maka objek tersebut dapat dideteksi dengan baik.</p>	

Gambar 12. Perbandingan Iterasi 1000 dan iterasi 2000.

4. Evaluasi Sistem

Dapat dilihat bahwa banyak sekali perbedaan ketika kita menggunakan sistem dengan data training yang menggunakan 1000 iterasi atau dengan menggunakan sistem dengan data training yang menggunakan 2000 iterasi. Dari hasil komparasi diatas dapat diambil kesimpulan bahwa semakin tinggi proses training pada sistem maka hasil pendeteksian yang dihasilkan akan semakin baik pula. Berikut perbedaan tingkat akurasi sistem dengan menggunakan 1000 iterasi maupun menggunakan 2000 iterasi:

Tingkat Akurasi Sistem 1000 Iterasi	Tingkat Akurasi Sistem 2000 Iterasi
$\text{Tingkat Akurasi Sistem} = \frac{\text{Jumlah data testing terdeteksi}}{\text{Total jumlah data testing}}$	$\text{Tingkat Akurasi Sistem} = \frac{\text{Jumlah data testing terdeteksi}}{\text{Total jumlah data testing}}$
$= \frac{51}{60} \times 100$	$= \frac{55}{60} \times 100$
$\text{Tingkat Akurasi Sistem} = 0,85 \times 100 = 85\%$	$\text{Tingkat Akurasi Sistem} = 0,91 \times 100 = 91\%$

Gambar 13. Perbandingan tingkat akurasi sistem.

Sehingga dapat diketahui dari 60 data testing yang ada keberhasilan sistem mendeteksi objek sebesar 91%. Dengan kesalahan 9% dimana sistem tidak dapat mendeteksi objek dengan warna baju berbeda, objek yang melakukan aktivitas yang berbeda, objek yang jauh dari jangkauan CCTV, dan objek yang terkena noise seperti halangan kayu dan objek lain yang menutupi.

V. KESIMPULAN

Sistem dapat mendeteksi objek (orang) yang berbeda-beda dengan menggunakan 2

data training yaitu data training dengan 1000 iterasi dan data training dengan 2000 iterasi. Hasil yang didapatkan setelah melakukan implementasi terhadap sistem bahwa dengan menggunakan data training 2000 iterasi memperoleh tingkat akurasi sistem sebesar 91% sedangkan dengan menggunakan data training 1000 iterasi hanya memperoleh tingkat akurasi sistem sebesar 85%. Maka dapat disimpulkan bahwa semakin besar iterasi data training yang kita gunakan, maka hasil yang diperoleh juga semakin maksimal.

VI. DAFTAR PUSTAKA

- [1] A. A. Pradana, C. Casman, and N. Nur'aini, "Pengaruh Kebijakan Social Distancing pada Wabah COVID-19 terhadap Kelompok Rentan di Indonesia," *J. Kebijak. Kesehat. Indones. JKKI*, vol. 9, no. 2, pp. 61–67, 2020, [Online]. Available: <https://jurnal.ugm.ac.id/jkki/article/view/55575>.
- [2] N. A. Pratama and D. Hidayat, "Pengetahuan dan Perilaku Masyarakat Memaknai Social Distancing," *J. Digit. Media Relatsh.*, vol. 2, no. 1, pp. 1–10, 2020.
- [3] A. Santoso and G. Ariyanto, "Implementasi Deep Learning Berbasis Keras Untuk Pengenalan Wajah," *Emit. J. Tek. Elektro*, vol. 18, no. 01, pp. 15–21, 2018, doi: 10.23917/emitor.v18i01.6235.
- [4] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016-Decem, pp. 779–788, 2016, doi: 10.1109/CVPR.2016.91.