

# Identifikasi Garis Telapak Tangan Dengan Metode *MobileNet Convolutional Neural Network* Untuk Sistem Presensi Siswa

Muhammad Hamdi Sukriyandi<sup>1</sup>, Achmad Solichin<sup>2</sup>  
<sup>1,2</sup>Fakultas Teknologi Informasi, Universitas Budi Luhur, Indonesia

---

## Article Info

### Article history:

Received Nov 22, 2022

Revised Feb 07, 2023

Accepted Feb 16, 2023

---

### Keywords:

Palm Print

Biometric

Deep learning

Data augmentation

Convolutional neural network

---

## ABSTRACT

The attendance system currently used at SMK Taruna Terpadu 1, which has nine majors and around 5.000 students, is done manually. This leads to many statistics being recorded by hand, making it difficult to keep track of and find when needed. To solve this problem, attendance recording is expected to be digitized using biometrics, which are computer-based recognition systems that recognize physical traits. Biometrics can come in the form of faces, irises, fingerprints, and handprints. However, contact fingerprinting will be unavailable during the COVID-19 pandemic, and facial recognition poses challenges such as skin color, mask use, and identical twins. Therefore, fingerprint biometrics and palm features are more attractive options for contactless human identification technology, as they are more accurate, reliable, and secure. This article proposes the use of MobileNetV2's augmented facts, ROI detection, and pre-trained convolutional neural network (CNN) models to identify palm features and lighting fixtures. The authors used data augmentation, ROI detection, and identification with the pre-trained MobileNetV2 model to test the dataset from SMK Taruna Terpadu 1 and achieved an accuracy result of up to 99.98%.

Copyright © 2023 Universitas Indraprasta PGRI.  
All rights reserved.

---

## Corresponding Author:

Muhammad Hamdi Sukriyandi,  
Magister Ilmu Komputer,  
Universitas Budi Luhur,  
Jl. Ciledug Raya, Pesanggrahan, Jakarta Selatan.  
Email: hamdi.sukriyandi@gmail.com

---

## 1. PENDAHULUAN

Sebelumnya, pencatatan kehadiran di SMK Taruna Terpadu 1 dilakukan secara manual. Namun, dengan perkembangan teknologi, digitalisasi pencatatan kehadiran bisa dilakukan dengan beberapa metode, salah satunya dengan biometric. Biometric sebagai teknologi untuk mengidentifikasi dengan ciri biologis secara digital dapat menggantikan kartu atau tanda pengenalan lainnya. Teknologi biometric bisa berupa fisik seperti wajah, iris, sidik jari, dan garis telapak tangan. Pencatatan kehadiran yang baik sangat penting untuk dilakukan guna mengetahui dan mengontrol peserta didik di SMK Taruna Terpadu 1 yang telah menjadi sekolah Kluster dan Sekolah Rujukan di Kabupaten Bogor. Demi mencapai mutu pendidikan yang berkualitas, semua sumberdaya kependidikan harus didukung, mulai dari aspek manajemen, kurikulum, kesiswaan, pengembangan fasilitas, personalia (SDM), dan aspek keuangan.

Namun saat pandemi Covid-19 dibutuhkan metode yang menghindari sentuhan dan menaati protokol kesehatan. Dilansir dari Tempo.co kasus Covid-19 dari *cluster* sekolah di daerah bogor mengalami peningkatan dua kali lipat dalam sehari. Beberapa waktu lalu tepatnya pada Senin, 31 Januari 2022 Dinas Kesehatan Kota Bogor mencatat terdapat 45 orang dinyatakan positif Covid-19 lalu kemudian pada catatan Selasa, 1 Februari 2022 menjadi 85 orang yang semuanya adalah siswa dan guru. Dengan adanya kasus tersebut Dinas Kesehatan Kota Bogor menyatakan akan menghentikan PTM atau pembelajaran tatap muka di sekolah yang siswa maupun guru terpapar Covid-19. Sekolah akan ditutup selama 5 hari dan akan ditambah menjadi 14 hari bila

ada peningkatan kasus Covid-19 di sekolah tersebut [1]. Sementara di DKI dilansir dari Kompas.com kasus Covid-19 yang dicatat Pemerintah Provinsi DKI terdapat 90 sekolah yang ditutup akibat adanya kasus Covid-19. Selanjutnya di Depok tercatat terdapat 239 kasus Covid-19 dari lingkungan sekolah, yang mengakibatkan 34 sekolah menghentikan kegiatan PTM 100 persen [2]. Oleh karena itu, sidik jari tidak dapat digunakan sebagai metode presensi karena harus bersentuhan dengan alat, kemudian ada tantangan dalam pengenalan wajah yang harus memakai masker dan ada kemungkinan bias rasial [3].

Kebijakan pengenalan wajah memiliki banyak kekhawatiran publik mengenai penyalahgunaan privasi dan bias rasial. Mengingat situasi di mana pengenalan wajah digunakan, potensi bahaya dari ketidakakuratan atau penyalahgunaan terlalu signifikan untuk diabaikan. Sistem berbasis data ini memiliki beberapa kelemahan termasuk potensi pelanggaran privasi, pengawasan, hasil yang tidak konsisten berdasarkan warna kulit dan profil ras [3].

Biometrik garis telapak pada tangan adalah alternatif yang menarik untuk teknologi pengenalan wajah, yang diharapkan lebih akurat, andal, dan aman di dunia nirsentuh baru. Telapak tangan manusia mengandung banyak fitur yang dapat diidentifikasi, seperti garis yang khas dan lipatan kulit. Untuk alasan ini, biometrik telapak tangan dapat secara akurat mengidentifikasi orang dari semua ras dan warna kulit, dan bahkan dapat membedakan antara kembar identik, dan bahkan ada penelitian terkait menggunakan citra telapak tangan untuk klasifikasi gaya belajar [4], tetapi tantangan dalam mendeteksi fitur telapak tangan ini adalah deteksi tanpa sentuhan diperlukan untuk mencapai akurasi pengenalan maksimum dan meminimalkan tingkat kesalahan. Namun, saat melakukan pra-introduksi atau mengidentifikasi garis halus pada telapak tangan, deteksi garis halus pada telapak tangan diperlukan terlebih dahulu karena garis halus pada telapak tangan yang teridentifikasi memiliki latar belakang dan pencahayaan yang berbeda. Hal ini juga membutuhkan arsitektur *Convolutional Neural Network* dengan sumber daya yang kecil, yakni sebuah perangkat yang memiliki spesifikasi yang menengah.

Pada penelitian terdahulu penerapan *Convolutional Neural Network* dalam MobileNetV2 dan *Support Vector Machine* (SVM) diterapkan pada dataset PolyU yang sudah dalam bentuk kotak yang sekaligus merupakan *Region of Interest* (ROI) [5]. Dengan data tersebut proses klasifikasi sudah bisa langsung dilakukan namun bagaimana jika diterapkan pada data yang diambil melalui telepon seluler dengan latar belakang dan pencahayaan yang berbeda – beda, sudah pasti dibutuhkan metode untuk mendeteksi ROI garis telapak tangan terlebih dahulu. Lain halnya dengan penelitian yang meneliti verifikasi telapak tangan pada telepon pintar penelitiannya menerapkan pendeteksian ROI terlebih dahulu sebelum melakukan klasifikasi dan menghasilkan akurasi yang baik [6]. Namun apakah metode tersebut bisa diterapkan pada dataset telapak tangan yang akan digunakan pada sistem presensi ini dengan hasil akurasi yang baik juga.

Berdasarkan pengamatan di atas, penulis menganjurkan pemanfaatan teknik untuk menangkap atau memilih fitur dengan terlebih dahulu memperluas catatan pelatihan dengan cara melakukan augmentasi data atau memperbanyak data dari data yang ada guna menghasilkan model yang sesuai. Kemudian, langkah selanjutnya adalah mengekstrak garis halus pada telapak tangan dari foto yang diambil. Dianotasi dengan tanda tangan dengan mengklasifikasikan penggunaan *Tiny Yolo* dan mengklasifikasikan dengan penggunaan struktur *MobileNet Convolutional Neural Network*. Ini merupakan arsitektur CNN yang ringan dan dapat mengimplementasikan faktor serta konvolusi yang dalam dan sangat cocok untuk digunakan pada perangkat seperti gawai atau *smartphone*.

Hasil dari pengamatan ini diharapkan dapat memungkinkan peneliti untuk membuat versi yang mengidentifikasi garis di telapak tangan tanpa menyentuh latar belakang atau pencahayaan yang berbeda. Versi yang dihasilkan dapat digunakan sebagai versi untuk mengetahui jejak atau garis halus pada telapak tangan siswa dalam sistem kehadiran atau presensi para siswa di SMK Taruna Terpadu 1 Bogor. Dengan adanya sistem ini, akan menjadi salah satu penerapan protokol kesehatan yang paling efektif dan paling efisien dalam kegiatan belajar dan mengajar tatap muka di berbagai lembaga pendidikan dengan jumlah siswanya yang cukup banyak.

Tabel 1. Penelitian Terkait

Paper	Metode	Dataset	Hasil
[7]	<i>Similarity metric hashing network</i>	<i>PolyU</i>	Setelah percobaan ekstensif dengan tiga basis data sidik jari referensi, ditemukan hasil bahwa model tersebut dapat mencapai akurasi kompetitif hingga 99,30%.
[6]	<i>DeepMPV+</i>	<i>MPD (Tongji Mobile Palmprint Dataset)</i>	Pada penelitian ini, akurasi terbaik yang dicapai adalah 99,18% dengan menerapkan ekstraksi ROI palmprint atau pengenalan palmprint Tiny Yolo sebelum verifikasi palmprint tersebut.

[8]	<i>EE-PRnet</i>	<i>NTU Palmprint Database -IITD -Casia -Tongji PolyU</i>	Database baru yang disebut NTU Palmprints dari Internet (NTU-PI-v1), yang berisi 7.881 gambar dari 2.035 telapak tangan yang dikumpulkan dari Internet, memberikan hasil mencapai 99,61%.
[9]	<i>RSM</i>	<i>PolyU</i>	Metode ini mengandalkan 2DPCA untuk membuat subruang acak yang hampir tidak konsisten. Dari hasil pengujian tersebut dapat dicapai akurasi sampai dengan 99,97%.
[10]	<i>JDCFR</i>	<i>Hyperspectral</i>	Dengan <i>dataset hyperspectral palmprint</i> yang terdiri dari 53 band spektral dengan 110.770 citra telah dilakukan eksperimen dan mendapatkan hasil akurasi terbaik hingga 99.62%
[5]	<i>MobileNetV2 + SVM</i>	<i>PolyU</i>	Dengan menggantikan <i>softmax</i> dengan <i>SVM</i> pada arsitektur <i>MobileNetV2</i> akurasi terbaik yg didapat mencapai 100%
[11]	<i>Alexnet</i>	<i>PolyU</i>	Metode ini melakukan pemotongan terlebih dahulu ( <i>pre-crop</i> ) pada area ROI dan diterapkan pada basis data gambar telapak tangan <i>Palmprint Multi-Spektral PolyU</i> dan basis data cetak telapak tangan PolyU 2D+3D, dan tingkat pengenalan.
[12]	<i>Generative Adversarial Network (GAN)</i>	<i>IIT Delhi dan CASIA</i>	1,52% dan 0,37% <i>Equal Error Rates</i> (EER)
[13]	<i>Deep Scattering</i>	<i>PolyU</i>	Diuji pada database palmprint yang populer yakni <i>PolyU</i> dengan akurasi 99,4% dan 99,9% dengan memanfaatkan Pengklasifikasi Jarak Minimum dan SVM

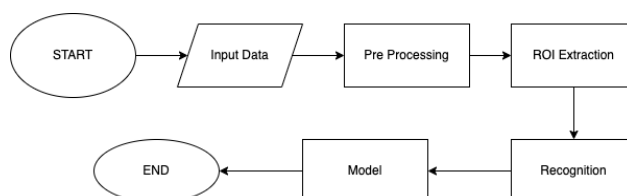
Dari ringkasan penelitian di atas, penulis menyoroti metode dari penelitian "*Towards Palmprint Verification on Smartphones*" yang memiliki akurasi terbaik sebesar 99,18% [5]. Berdasarkan penelitian tersebut, penulis mendapatkan ide untuk menerapkan deteksi ROI dengan menggunakan tiny YOLO untuk mendeteksi bagian ciri telapak tangan. Penulis juga mengambil ide dari penelitian "Mobilenet convolutional neural networks and support vector machines for palmprint recognition" yang memiliki akurasi terbaik sebesar 100% [4]. Berdasarkan penelitian tersebut, penulis mendapatkan ide untuk mengaplikasikan arsitektur tersebut untuk menyelesaikan tugas klasifikasi atau identifikasi dalam penelitian ini.

## 2. METODE

Pendekatan yang digunakan dalam penelitian ini adalah metode eksperimen. Tujuan dari metode eksperimen ini adalah untuk mencari nilai akurasi pada pengenalan ciri telapak tangan dari gambar telapak tangan menggunakan teknik *deep learning*. Dalam metode ini, gambar telapak tangan yang diambil menggunakan kamera smartphone dengan latar belakang dan pencahayaan yang berbeda-beda akan digunakan sebagai data latih dan data validasi. Proses pelatihan dan validasi ini akan menghasilkan model yang dapat digunakan untuk mengidentifikasi ciri telapak tangan. Pelatihan dan pengujian dilakukan di *google colabs notebook* dengan menggunakan bahasa pemrograman Python beserta *library* pendukungnya. Hasil dari penelitian ini diharapkan dapat menghasilkan akurasi yang tinggi dalam mengidentifikasi ciri telapak tangan dari gambar.

### 2.1. Rancangan Base Model

Berikut rancangan pada penelitian dalam mengenali ciri telapak tangan untuk diterapkan pada attendance system.



Gambar 1. Diagram alir pembuatan base model

Dari gambar 1 di atas ada beberapa tahapan pada rancangan tersebut berikut penjelasannya:

1. Masukkan gambar atau gambar telapak tangan yang diambil dengan kamera *smartphone*.
2. Pre-processing atau Pra-pemrosesan dilakukan untuk mengubah ukuran gambar sebelum menganotasi atau memberi label, pelabelan atau anotasi telapak tangan, untuk sebelum dilakukan *training ROI detection*. Karena *deep learning* membutuhkan data dalam jumlah besar, dilakukan *pre-processing* untuk meningkatkan data melalui teknik augmentasi data seperti transformasi kecerahan, kontras, rotasi, dan penskalaan. Hal ini bertujuan untuk meningkatkan kinerja model deteksi ROI.

3. ROI *extraction*, yakni terlebih dahulu melakukan deteksi ROI (*Region of Interest*) dari citra telapak tangan dengan uji coba menggunakan Tiny Yolo 4. Dari hasil deteksi ROI, area citra di dalam bounding box tersebut akan diekstraksi menjadi file gambar tersendiri dengan tiga keluaran tipe gambar yakni RGB, *grayscale* juga adaptif *threshold*.
4. Saat menerapkan algoritma klasifikasi, peneliti menggunakan algoritma *convolutional neural network* CNN yang berbasis arsitektur *mobileNetV2* dengan *softmax classifier* dan SVM. Latih model dengan set data yang telah diproses sebelumnya seperti augmentasi data. Waktu dan input yang berbeda digunakan di sini untuk pengujian.
5. Simpan model untuk implementasi pada prototipe sistem presensi.

## 2.2. Data Set

Foto yang diperoleh dari 30 subjek masing-masing 5 foto telapak tangan kanan dan 5 foto telapak tangan kiri sehingga berjumlah total 300 foto. Setelah itu file foto tersebut di-*resize* menjadi 512 x 683 untuk memperkecil file size, seperti yang disarankan oleh keras sebagai salah satu tools dalam computer vision, Saat melatih model vision, dilakukan penngubaha ukuran gambar asli menjadi gambar dengan dimensi yang lebih rendah ditujukan untuk memungkinkan pembelajaran mini-batch dan juga untuk menjaga komputasi agar tetap rendah [14].

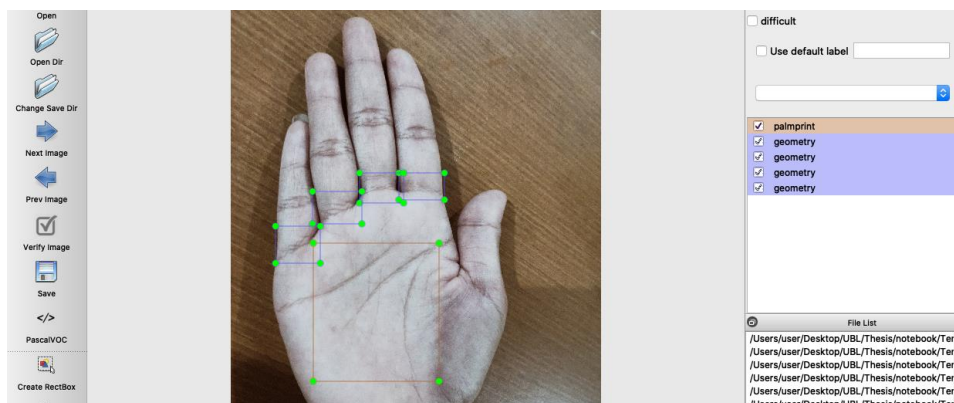
Pada uji coba ini diambil satu dari masing – masing foto tangan kanan dan kiri dari 30 subjek yaitu total 60 foto gambar yang akan dilakukan preprocessing untuk tahapan selanjutnya yaitu pelabelan *dataset* untuk kebutuhan training deteksi ROI dari telapak tangan dengan YOLOv4.



Gambar 2. Data set foto telapak tangan

## 2.2. Anotasi

Anotasi dataset atau melabeli foto dengan kelas yang merupakan ciri dari telapak tangan. Kelas yang terdapat pada tahap ini ada dua kelas yaitu *geometry* dan *palmprint* (beberapa ciri yang terdapat dalam satu area). Proses ini dilakukan dengan menggunakan tools yaitu *labelimg*.



Gambar 3. Proses anotasi dengan *tools labeling*

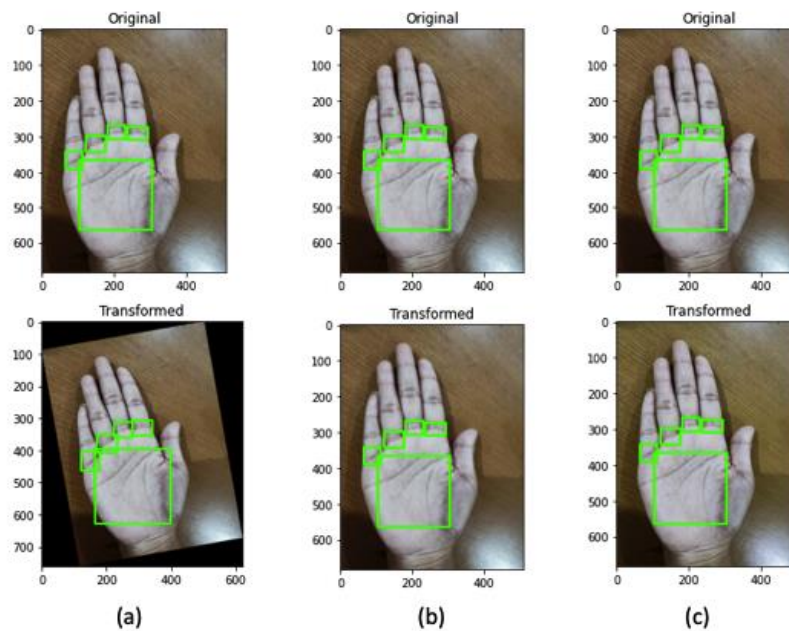


### 2.3. Augmentasi Data

Sebelumnya alasan kenapa dari 300 foto yang diberi label hanya 60 adalah untuk mempersingkat waktu dalam proses pelabelan. Dan untuk memenuhi kebutuhan dataset yang banyak maka akan dilakukan augmentasi dataset yang sudah diberi label. Teknik augmentasi yang dilakukan pada ujicoba kali seperti pada gambar 4 ini meliputi (a) rotasi gambar, (b) *average blur*, dan (c) *raise hue*. Setelah dilakukan augmentasi maka akan menghasilkan 2280 file foto dengan ekstensi .jpg dan 2280 file label dengan ekstensi .txt dengan detail augmentation yang dilakukan seperti pada tabel 2.

Tabel 2. Teknik Data Augmentasi

No	Teknik	Jumlah Eksekusi
1	Rotasi setiap 10 derajat	35
2	<i>Average Blur</i>	1
3	<i>Raise Hue</i>	1
4	Normal	1

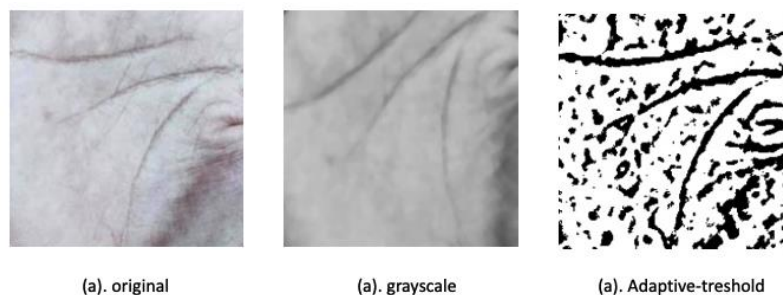


Gambar 4. Hasil augmentasi Rotasi, *average blur*, dan *raise hue*

## 3. HASIL DAN PEMBAHASAN

### 3.1. Hasil Ekstraksi ROI

Dari hasil deteksi itu dilakukan ekstraksi dengan menggunakan dataset yang sudah dikelompokkan dalam folder kelas masing. Pada uji coba ini ada 60 kelas dan berisikan gambar yang sudah diaugmentasi berjumlah 160 foto per kelas sehingga berjumlah 9600 foto. *bounding box* yang terprediksi sebagai *palmprint* kemudian di *crop* dan *resize* dan disimpan dengan tiga variasi gambar, yaitu gambar dengan warna *original*, *grayscale* dan adaptif *threshold* dengan ukuran yang telah di-*resize* menjadi 224 x 224 yang hasilnya terlihat seperti gambar 5.

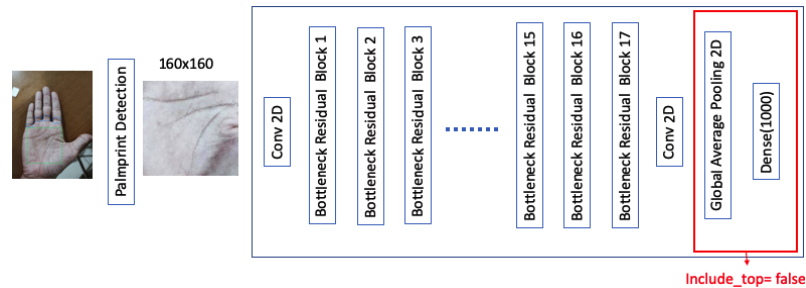


Gambar 5. Hasil ekstraksi dengan tiga varian gambar

### 3.2. Klasifikasi Ciri Telapak Tangan

Dari hasil ekstraksi telah didapatkan tiga varian gambar yang akan menjadi masukan untuk proses *training* masing – masing varian dengan skenario uji coba.

Sebelum melakukan proses *training* terlebih dahulu penulis membagi data menjadi 3 bagian yaitu 70% data latih, 20 % data validasi untuk mengetahui tred pada saat pelatihan, dan 10% data test [15] [16] dari total 9600 data. Dan kemudian data tersebut diaugmentasi sesuai skenario uji coba.

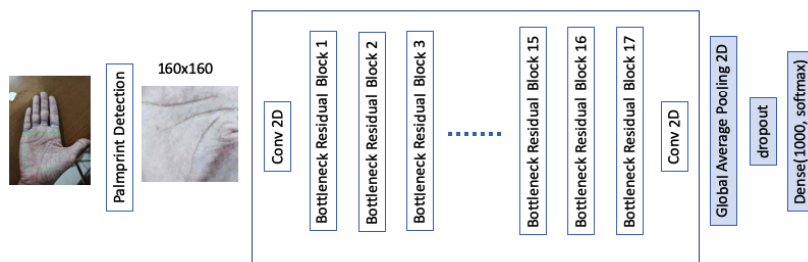


Gambar 6. Pretrained model MobileNet v2

#### MobileNet V2 Softmax

Dengan menggunakan *pretrained model* mobilenetv2 dan dengan menggunakan *activation function softmax* berikut di bawah ini formula *loss function* yang terdapat pada *activation function softmax*[17], akurasi pada *training* pertama dengan *epoch* sebesar lima masing – masing varian mendapatkan akurasi, *original* 92.80%, *grayscale* 91.46%, dan *adaptive-threshold* 72.74%. dengan input size 160 x160 dengan 3 kanal. Dengan dilakukan perubahan dimensi gambar dapat memperkacil komputasi saat *training*[14] .

$$L_i = -\log(e^{S_j} / \sum_j e^{S_j}), L = \frac{1}{N} \sum_{i=1}^N L_i$$



Gambar 7. Mobilenetv2 layer dengan *activation function softmax*

Dari model di atas dilakukan fine tuning dengan ikut melatih 54-layer terakhir *pretrained model* dengan penambahan *epoch* sebesar 100 jadi total dari *epoch* pertama dan *epoch fine tuning* menjadi 105, dan *training* akan start dari *epoch* ke 5 hingga 105, alasan memilih menambahkan 100 *epoch* pada *fine tuning* adalah untuk menguji coba karena disini penulis melatih 54-layer terakhir yang pada awal *start* mendapatkan hasil akurasi dibawah 90% dan akurasi validasi dibawah 1%. Dan dengan epoch tersebut masing – masing varian mendapatkan akurasi, *original* 99.98%, *grayscale* 99.97%, dan *adaptive-threshold* 99.94%.

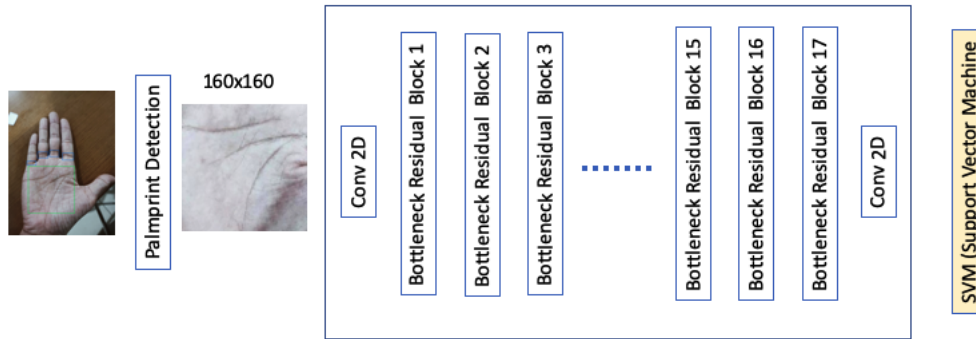
Tabel 3. Hasil *training* MobileNet v2 Softmax

No	Varian Gambar	Augmentasi	Total Epoch	Akurasi (%)
1	<i>Original(rgb)</i>	<i>rescale</i> =1. / 255, <i>rotation_range</i> =40, <i>shear_range</i> =0.2, <i>zoom_range</i> =0.2	105	99,98
2	<i>Grayscale</i>	<i>rescale</i> =1. / 255, <i>rotation_range</i> =40, <i>shear_range</i> =0.2, <i>zoom_range</i> =0.2	105	99,97
3	<i>Adaptive-threshold</i>	<i>rescale</i> =1. / 255, <i>rotation_range</i> =40, <i>shear_range</i> =0.2, <i>zoom_range</i> =0.2	105	99,94

**Mobilenet V2 dengan SVM**

Pada skenario ini uji coba dilakukan dengan mengambil *pretrained model Mobilenetv2* tanpa dua *layer* terakhir dan mengunci semua *layer* menjadi *trainable false* kemudian menjadikannya sebagai *feature extractor* seperti pada gambar 8 dan menjadi *input data training* untuk *training Support Vector Machine (SVM)* berikut di bawah ini formula *loss function* yang terdapat pada *activation function SVM*[17]. Pada skenario ini dilakukan *training* sekali jalan sebanyak total data *training*.

$$L_i = \sum_{j \neq y_i} \max(0, s_j - s_{y_i} + 1), L = \frac{1}{N} \sum_{i=1}^N L_i$$



Gambar 8. Layer Mobilenet V2 sebagai *feature extractor* untuk SVM

Tabel 4. Hasil training MobileNet v2 dengan SVM

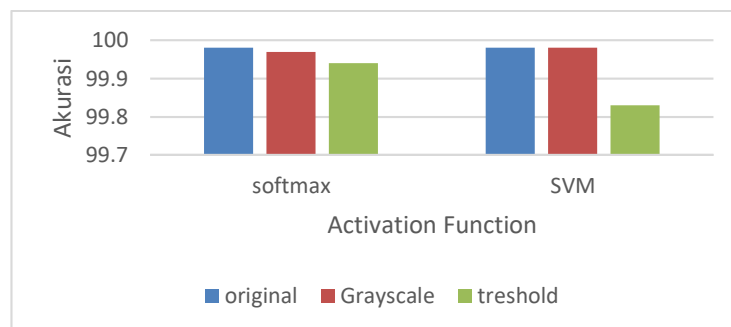
No	Varian Gambar	Augmentasi	Data Train/Test	Akurasi (%)
1	<i>Original(rgb)</i>	<i>rescale=1. / 255, rotation_range=40, shear_range=0.2, zoom_range=0.2</i>	6720/1920	99,98
2	<i>Grayscale</i>	<i>rescale=1. / 255, rotation_range=40, shear_range=0.2, zoom_range=0.2</i>	6720/1920	99,98
3	<i>Adaptive-threshold</i>	<i>rescale=1. / 255, rotation_range=40, shear_range=0.2, zoom_range=0.2</i>	6720/1920	99,83

**3.3. Perbandingan Softmax dan SVM**

Dari hasil uji coba yang dilakukan perbedaan diantaranya tidak signifikan hampir sama namun dilihat dari ukuran atau *size* dari model itu sendiri sangat terlihat berbeda jauh sekali, model dengan *mobilenetv2-softmax* mempunyai *size* 14.3 MB sedangkan *mobilenetv2* dengan SVM 1.54 GB. Ukuran tersebut sekitar 100 kali lebih besar dari *softmax*.

Tabel 5. Perbandingan MobileNet v2 dengan Softmax dan SVM

No	Varian Gambar	Classifier	Size	Akurasi (%)
1	<i>Original(rgb)</i>	<i>Mobilenet V2 + Softmax</i>	14.3 MB	99,98
2	<i>Grayscale</i>	<i>Mobilenet V2 + Softmax</i>	14.3 MB	99,97
3	<i>Adaptive-threshold</i>	<i>Mobilenet V2 + Softmax</i>	14.3 MB	99,94
4	<i>Original(rgb)</i>	<i>Mobilenet V2 + SVM</i>	1.54 GB	99,98
5	<i>Grayscale</i>	<i>Mobilenet V2 + SVM</i>	1.54 GB	99,98
6	<i>Adaptive-threshold</i>	<i>Mobilenet V2 + SVM</i>	1.54 GB	99,83



Gambar 9. Grafik perbandingan akurasi model

Dari evaluasi model di atas model yang akan diterapkan pada prototipe system presensi adalah model dengan *pretrained model* mobilenetV2 dengan *classifier softmax* dengan inputan gambar berwarna, yang diharapkan dapat diterapkan dengan baik pada system presensi siswa.

### 3.4. Pembahasan

Berdasarkan hasil uji coba yang telah diuraikan di atas bahwa dengan penggunaan augmentasi data bisa dapat membantu meningkatkan akurasi dalam *object detection* dan juga dalam proses klasifikasi. Mengingat jumlah dataset yang diperoleh tidak terlalu banyak dan data foto yang diambil tidak mempunyai banyak variasi dalam hal transformasi seperti kemiringan, jarak, dan pencahayaan, bahkan bisa dikatakan foto yang didapat hampir sama dengan hanya sedikit perbedaan pergeseran posisi letak tangan.

Kemudian dengan menggunakan tiny YOLO4 dapat mengekstraksi ROI telapak tangan dengan akurasi yang baik. Dengan data augmentasi berupa rotasi 10 derajat sebanyak 35 kali hingga membentuk lingkaran sebelum pelatihan object detection pada yolo yang hasilnya dapat mendeteksi telapak tangan dengan kemiringan yang beragam. Berbeda sekali dengan hasil deteksi dengan tanpa data augmentasi model tidak dapat mendeteksi telapak tangan yang posisinya agak miring melainkan harus tegak. Setelah diuji coba dengan gambar telapak yang miring, model yang dilatih dengan data augmentation dapat dengan mudah mengenali ROI dari telapak tangan. Hasil dari ROI *detection* itu sendiri di-*crop* dan disimpan berupa file gambar dengan ekstensi jpg.

Dengan menggunakan metode *object detection*, tugas untuk mengidentifikasi telapak tangan tanpa sentuhan dengan menggunakan kamera menjadi sangat memungkinkan karena metode tersebut bisa memisahkan area gambar yang dibutuhkan atau ROI dari latar belakang yang berbeda – beda. Dari hasil pemisahan ROI, penulis mencoba menyimpannya dengan tiga tipe gambar, yang pertama gambar asli langsung crop, kemudian yang kedua gambar dengan warna grayscale, dan yang terakhir tipe gambar yang diubah dengan *adaptive-threshold*. Semua disimpan dengan jumlah yang sama, dengan tujuan untuk diperbandingkan hasil training dari masing – masing tipe gambar.

Pada bagian training untuk klasifikasi penulis mentraining masing – masing tipe gambar dengan pretrained model yakni mobilenetV2 semua diberlakukan sama dengan parameter yang sama, data augmentasi yang sama, namun hasilnya berbeda – beda walau hanya sedikit sekali selisihnya antara satu tipe dengan tipe yang lain. *Classifier* yang digunakan yakni softmax dan *support vector machine*, dengan menggunakan dua *classifier* tersebut total menjadi 6 skenario. Dari hasil percobaan antara dua *classifier* tersebut hasil akurasi tidak jauh berbeda namun yang menjadi perbedaan yang sangat terlihat adalah sizenya. Ukuran model dari *support vector machine* lebih besar dari *softmax*.

Dengan model yang telah dilatih penulis memilih model yang menggunakan input gambar dengan tipe normal tanpa dirubah menjadi grayscale maupun *adaptive-threshold* dengan *classifier softmax*. Dalam penerapannya pada prototipe model tersebut menjadi model dasar untuk dijadikan model baru dengan jumlah kelas yang berbeda, yakni kelas baru yang akan ditambahkan. Dalam proses penambahan data baru direkam dengan menggunakan object detection kemudian ROI disimpan pada direktori local dengan ekstensi jpg sebanyak 50 gambar yang akan disebar ke direktori train, test, dan validation masing – masing sebanyak 35 data train, 10 data *validation*, dan 5 data *test*.

Berebak *base model* tersebut penulis membuat model baru dengan mengganti layer terakhir classifier dengan *classifier* yang baru sesuai dengan jumlah kelas yang akan ditraining. pada penelitian ini base model sudah mentraining 60 kelas, jadi selanjutnya untuk menambah satu telapak tangan lagi *classifier* diubah menjadi 61 begitu seterusnya jika tambahnya dua telapak tangan makan jumlah kelas yang baru 62. Dengan base model yang sudah ditraining sebelumnya hasil training model yang baru tersebut bisa mendapatkan akurasi terbaik hingga 100% hanya sampai dengan 10 epoch.



Proses *training* ulang atau pendaftaran telapak tangan baru dilakukan pada prototipe system presensi. Pada praktiknya telapak tangan yang belum ditraining diharuskan mendaftar dengan mengisi formulir dan lalu melakukan pengambilan telapak tangan. Pada saat pengambilan gambar telapak tangan ROI akan terdeteksi dan akan tersimpan sebanyak 50 gambar mengikuti pergerakan gambar diikuti dengan adanya *bounding box* biru yang bertuliskan jumlah foto hingga bertuliskan sudah terdaftar saat pengambilan gambar sudah mencapai 50, barulah setelahnya bisa dilakukan training ulang dengan mennggunakan data baru berikut juga data lama. Hasil dari training ulang tersebut model baru akan disimpan pada direktori local bernama model, dan memperbaharui *active-model.txt* dengan nama model yang baru secara otomatis, dan system presensi pun siap digunakan dengan telapak tangan yang baru.

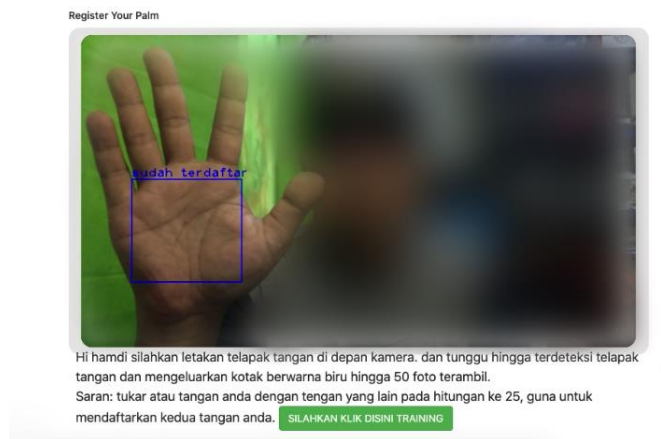
**3.5. Prototipe**

Untuk melakukan uji coba pada prototipe kali ini dilakukan pendaftaran telapak tangan sebagai input untuk nantinya dilatih dengan model yang sudah dibuat. Proses pendaftaran dilakukan dengan menangkap gambar secara langsung dari kamera *webcam* dengan menggunakan *object detection model yolo* yang sudah dilatih untuk mengekstraksi ROI telapak tangannya. Telapak tangan yang ditangkap diambil sebanyak 50 gambar dengan ukuran 160x160.

Pada proses pengambilan gambar untuk *training* di prototipe ini ada ketentuan yang harus dipenuhi seperti jarak dan minimum width dan height dari bounding box hingga *behavior* telapak tangan saat melakukan pengambilan gambar seperti pada tabel 6 di bawah ini.

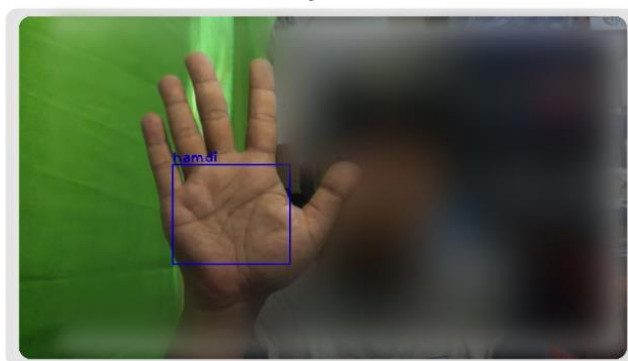
Tabel 6. Perbandingan MobileNet v2 dengan Softmax dan SVM

No	Parameter	Nilai
1	Jarak telapak tangan dengan <i>webcam</i>	± 20 cm
2	<i>Minimum width bounding box</i>	320 px
3	<i>Minimum height bounding box</i>	320 px
4	Menggeser tangan ke kanan dan kiri perlahan	Y
5	Merotasi tangan secara drastis	N
6	Menggunakan cahaya tambahan	Y



Gambar 10. Proses pendaftaran telapak tangan untuk training

### Palm Print Attendance System



Mulai absensi:  On/Off  Bila belum terdaftar silahkan klik tombol daftar ini [daftar](#)

Gambar 11. Hasil identifikasi dengan model yang telah dilatih

#### 4. PENUTUP

Dari hasil uji coba dengan beberapa skenario metode yang diteliti dihasilkan beberapa kesimpulan yaitu dengan menggunakan augmentasi data dapat menambah akurasi pada deteksi ROI telapak tangan juga berdampak baik pada proses pelatihan model untuk klasifikasi yang akan menambah referensi bagi data set yang sedikit.

*Pretrained model mobilenetv2* dengan *classifier softmax* menghasilkan model yang lebih kecil ukurannya bila dibandingkan dengan SVM (*Support Vector Machine*) yang menghasilkan ukuran model yang sangat besar walaupun hasil akurasi dari model SVM tidak kalah baik dari softmax.

Pada penerapannya model dengan *mobilenetv2 network* dirasa mempunyai kekurangan saat menambahkan kelas baru, dengan adanya penambahan data gambar seiring dengan bertambahnya kelas atau subjek baru maka akan dibutuhkan kapasitas *storage* yang besar dan harus di-*training* ulang dengan waktu yang cukup lama.

#### DAFTAR PUSTAKA

- [1] Antara, "Dalam Sehari, Kasus Covid-19 di Sekolah Kota Bogor Meningkat 2 Kali Lipat," *Metro Tempo.co*, 2022.
- [2] M. A. Hapsari, "Muncul Klaster Penularan Covid-19 di Sekolah, Saatnya PTM Dihentikan Sementara," *Kompas.com*, 2022.
- [3] L. Kontsevich, "The future of biometrics is in the palm of your hand | Biometric Update," *biometricupdate.com*, 2020.
- [4] Irennada, A. Solichin, and G. Brotosaputro, "Klasifikasi Gaya Belajar Mahasiswa Berdasarkan Garis Telapak Tangan Menggunakan Convolutional Neural Network," *Jurnal Nasional Pendidikan Teknik Informatika: JANAPATI*, vol. 11, no. 3, pp. 269–279, Dec. 2022.
- [5] A. Michele, V. Colin, and D. D. Santika, "MobileNet Convolutional Neural Networks and Support Vector Machines for Palmprint Recognition," *Procedia Comput Sci*, vol. 157, pp. 110–117, Jan. 2019.
- [6] Y. Zhang, L. Zhang, R. Zhang, S. Li, J. Li, and F. Huang, "Towards Palmprint Verification On Smartphones," Mar. 2020.
- [7] C. Liu, D. Zhong, and H. Shao, "Few-shot palmprint recognition based on similarity metric hashing network," *Neurocomputing*, vol. 456, pp. 540–549, Oct. 2021.
- [8] W. M. Matkowski, T. Chai, and A. W. K. Kong, "Palmprint Recognition in Uncontrolled and Uncooperative Environment," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 1601–1615, 2020.
- [9] I. Rida, R. Herault, G. L. Marcialis, and G. Gasso, "Palmprint recognition with an efficient data driven ensemble classifier," *Pattern Recognit Lett*, vol. 126, pp. 21–30, Sep. 2019.
- [10] S. Zhao, B. Zhang, and C. L. Philip Chen, "Joint deep convolutional feature representation for hyperspectral palmprint recognition," *Inf Sci (N Y)*, vol. 489, pp. 167–181, Jul. 2019.
- [11] W. Gong, X. Zhang, B. Deng, and X. Xu, "Palmprint recognition based on convolutional neural network-alexnet," *Proceedings of the 2019 Federated Conference on Computer Science and Information Systems, FedCSIS 2019*, pp. 313–316, Sep. 2019.
- [12] G. Wang, W. Kang, Q. Wu, Z. Wang, and J. Gao, "Generative Adversarial Network (GAN) Based Data Augmentation for Palmprint Recognition," *2018 Digital Image Computing: Techniques and Applications (DICTA)*, pp. 1–7, Dec. 2018.

- [13] S. Minaee and Y. Wang, "Palmpoint recognition using deep scattering network," *Proceedings - IEEE International Symposium on Circuits and Systems*, Sep. 2017.
- [14] "Learning to Resize in Computer Vision," *keras.io*, 2023.
- [15] F. Ramzan *et al.*, "A Deep Learning Approach for Automated Diagnosis and Multi-Class Classification of Alzheimer's Disease Stages Using Resting-State fMRI and Residual Neural Networks," *J Med Syst*, vol. 44, no. 2, pp. 1–16, Feb. 2020.
- [16] M. Elgendy, *Deep Learning for Vision Systems*. United States: Manning, 2020.
- [17] A. Rosebrock, *Deep Learning for Computer Vision with Python*, 3rd ed. pyimagesearch, 2019.