



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



RANCANG BANGUN SISTEM PENGHITUNG JUMLAH MANUSIA DAN PENGAWASAN DALAM RUANGAN

Sub Judul :

**Sistem Identifikasi Wajah Seseorang Dalam Ruangan
Menggunakan *Framework Tensorflow***

SKRIPSI

**POLITEKNIK
NEGERI
JAKARTA**

**Difna Yasmika
2203433006**

**PROGRAM STUDI INSTRUMENTASI KONTROL DAN
INDUSTRI**

JURUSAN TEKNIK ELEKTRO

POLITEKNIK NEGERI JAKARTA

2024



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



RANCANG BANGUN SITEM PENGHITUNG JUMLAH MANUSIA DAN PENGAWASAN DALAM RUANGAN

Sub Judul :

Sistem Identifikasi Wajah Seseorang Dalam Ruang
Menggunakan *Framework Tensorflow*

SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh gelar

Sarjana Terapan

POLITEKNIK
NEGERI
JAKARTA

Difna Yasmika

2203433006

PROGRAM STUDI INSTRUMENTASI KONTROL DAN
INDUSTRI

JURUSAN TEKNIK ELEKTRO

POLITEKNIK NEGERI JAKARTA

2024



© Hak Cipta milik Politeknik Negeri Jakarta

HALAMAN LEMBAR PERNYATAAN ORISINALITAS

Skripsi ini adalah hasil karya sendiri dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan dengan benar.

Nama : Difna Yasmika

NIM : 2203433006

Tanda Tangan :

Tanggal : 4 Februari 2024

POLITEKNIK
NEGERI
JAKARTA

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

LEMBAR PENGESAHAN SKRIPSI

Tugas Akhir diajukan oleh :

Nama : Difna Yasmika
NIM : 2203433006
Program Studi : Instrumentasi dan Kontrol Industri
Judul Tugas Akhir : Sistem Identifikasi Wajah Sescorang Dalam Ruangan Menggunakan *Framework Tensorflow*

Telah diuji oleh tim penguji dalam Sidang Tugas Akhir pada 5, Februari 2024 dan dinyatakan **LULUS**.

Pembimbing I : Hariyanto, S.Pd.,M,T
NIP. 199101282020121008

Depok, 12. Februari 2024

Disahkan oleh

Ketua Jurusan Teknik Elektro



Rika Novita Wardhani, S.T., M.T.

NIP. 197011142008122001



KATA PENGANTAR

Puji syukur saya panjatkan kepada Tuhan Yang Maha Esa, karena atas berkat dan rahmat-Nya, penulis dapat menyelesaikan Laporan Skripsi ini. Penulisan Skripsi ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana Terapan Politeknik Negeri Jakarta, Jurusan Teknik Elektro, Program Studi Instrumentasi dan Kontrol Industri. Skripsi ini berjudul “Rancang Bangun Sistem Penghitung Jumlah Manusia dan Pengawasan Dalam Ruangan” dengan sub judul “Sistem Identifikasi Wajah Seseorang Dalam Ruangan Menggunakan *Framework Tensorflow*”.

Penulis menyadari bahwa tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan samapi pada penyusunan tugas akhir ini, sangatlah sulit bagi penulis untuk menyelesaikan Tugas Akhir ini. Oleh karena itu, penulis mengucapkan terima kasih kepada:

1. Rika Novita Wardhani, S.T., M.T. selaku ketua Jurusan Teknik Elektro;
2. Sulis Setiowati, S.Pd., M.Eng, selaku Ketua Program Studi Instrumentasi dan Kontrol Industri
3. Hariyanto, S.Pd., M.T. selaku dosen pembimbing yang telah menyediakan waktu, tenaga, dan pikiran untuk mengarahkan penulis dalam penyusunan tugas akhir ini;
4. Orang tua dan keluarga penulis yang telah banyak memberikan bantuan dukungan material dan moral;
5. Wahyuningrum selaku teman satu tim dalam pengerjaan tugas akhir;
6. Teman-teman RPL IKI 2023 lain yang telah membantu dalam menyelesaikan skripsi ini; dan
7. Sahabat yang telah banyak membantu penulis dalam menyelesaikan tugas akhir ini.

Akhir kata, penulis berharap Tuhan Yang Maha Esa berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga Tugas akhir ini membawa manfaat bagi pengembangan ilmu.

Depok, Januari 2024

penulis

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Sistem Identifikasi Wajah Seseorang Dalam Ruangan Menggunakan Framework Tensorflow

ABSTRAK

Kegiatan absensi yang masih sering menggunakan kertas dianggap kurang efektif karena berpotensi hilang dan rusak, dalam proses absensi menggunakan metode face recognition hasil absensi dapat disimpan ke dalam folder khusus agar bukti kehadiran tidak dapat dimanipulasi. Selain itu banyaknya kejadian mahasiswa melakukan manipulasi absensi dapat dicegah dengan menambahkan perhitungan manusia untuk mendeteksi jumlah orang yang hadir secara fisik dalam ruangan sesuai dengan hasil absensi menggunakan face recognition. Berdasarkan permasalahan tersebut dibuatlah sistem facerecognition, dalam pembuatan sistem face recognition dibutuhkan beberapa library yang dipakai salah satunya adalah opencv dan tensorflow. Tensorflow dapat memungkinkan pembangunan model deep learning untuk memahami dan mengidentifikasi wajah dengan tingkat akurasi yang tinggi, salah satu algoritma deep learning yang digunakan yaitu adalah Convolutional Neural Network (CNN). Pada proses identifikasi wajah jarak antar kamera webcam yang dilakukan pada pagi hari dan malam hari dapat mempengaruhi hasil kinerja sistem. Dari setiap pengujian didapatkan jarak terbaik yaitu pada jarak 30cm dengan intensitas cahaya 36 lux dan 67 lux dengan hasil akurasi keberhasilan yang didapatkan yaitu 100%. Model identifikasi wajah yang dibuat menggunakan metode Convolutional Neural Network (CNN) sudah cukup bagus jika mendapatkan cahaya yang baik dan jarak yang sesuai.

Kata Kunci : *Raspberry Pi4 Model B, Webcam, Opencv, Tensorflow, Convolutional Neural Network*

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



ABSTRACT

Attendance activities that still often use paper are considered less effective because they have the potential to be lost and damaged, in the attendance process using the face recognition method the attendance results can be stored in a special folder so that proof of attendance cannot be manipulated. In addition, the many incidents of students manipulating attendance can be prevented by adding human calculations to detect the number of people physically present in the room according to the results of attendance using face recognition. Based on these problems, a face recognition system is made, in making a face recognition system, several libraries are needed, one of which is opencv and tensorflow. Tensorflow can enable the construction of deep learning models to understand and identify faces with a high level of accuracy, one of the deep learning algorithms used is the Convolutional Neural Network (CNN). In the face identification process, the distance between webcam cameras carried out in the morning and at night can affect the performance of the system. From each test, the best distance is obtained at a distance of 30cm with a light intensity of 36 lux and 67 lux with the results of the success accuracy obtained is 100%. The face identification model created using the Convolutional Neural Network (CNN) method is good enough if it gets good light and the appropriate distance.

Keywords : *Raspberry Pi4 Model B, Webcam, Opencv, Tensorflow, Convolutional Neural Network*

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



DAFTAR ISI

HALAMAN LEMBAR PERNYATAAN ORISINALITAS	iii
LEMBAR PENGESAHAN SKRIPSI.....	iv
KATA PENGANTAR.....	v
ABSTRAK	vi
ABSTRACT	vii
DAFTAR GAMBAR	x
DAFTAR TABEL	xi
BAB I PENDAHULUAN	13
1.1 Latar Belakang	13
1.2 Perumusan Masalah.....	14
1.3 Tujuan.....	15
1.4 Batasan Masalah.....	15
1.5 Luaran.....	15
BAB II TINJAUAN PUSTAKA	16
2.1. State Of The Art	16
2.2. Rapberry pi.....	19
2.3. Python.....	20
2.4. Open CV.....	21
2.5. Tensorflow	21
2.6. Keras.....	22
2.7. Artificial Intilegence (AI).....	22
2.8. Machine Learning	23
2.9. Deep Learning	24
2.10. CNN (<i>Convolution Neural Network</i>)	25
1. Convolutional Layers	25
2. Pooling Layer	25
3. Fully Connected Layer	26
2.11. MobileNetV2.....	26
2.12. Dataset.....	27
2.13. Webcam.....	28

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

2.14.	Layar Monitor	28
BAB III PERANCANGAN DAN REALISASI.....		30
3.1.	Perancangan Alat.....	30
3.1.1.	Deskripsi Alat.....	31
3.1.2.	Cara Kerja Alat.....	32
3.1.3.	Spesifikasi Alat	33
3.1.4.	Diagram Blok	35
3.2.	Realisasi Sistem Identifikasi Wajah	36
3.2.1.	Pengumpulan dan Pengkondisian Dataset.....	38
3.2.2.	Proses Pembuatan Model Identifikasi Wajah.....	41
3.2.3.	Realisasi Alat Identifikasi Wajah	45
3.2.4.	Proses Penyimpanan Identifikasi Wajah	46
BAB IV PEMBAHASAN		48
4.1	Pengujian Identifikasi Wajah Pada Pagi Hari	48
4.1.1	Deskripsi Pengujian.....	49
4.1.2	Daftar Alat dan Bahan.....	49
4.1.3	Prosedur Pengujian.....	49
4.1.4	Analisa Data Hasil Pengujian Pagi Hari.....	50
4.2	Pengujian Identifikasi Wajah Pada Malam Hari	53
4.2.1	Deskripsi Pengujian.....	54
4.2.2	Daftar Alat dan Bahan.....	54
4.2.3	Prosedur Pengujian.....	54
4.2.4	Analisa Data Hasil Pengujian.....	55
BAB V KESIMPULAN		58
5.1.	Kesimpulan.....	58
5.2.	Saran.....	59
DAFTAR PUSTAKA.....		60
LAMPIRAN		63



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

DAFTAR GAMBAR

Gambar 2. 1 Raspberry Pi 4B.....	19
Gambar 2. 2 Artificial Intelligence.....	22
Gambar 2. 3 Proses Machine Learning	23
Gambar 2. 4 Deep Learning	24
Gambar 2. 5 Proses Convolution Layer	25
Gambar 2. 6 Operasi Convolution.....	26
Gambar 2. 7 Fully Connected Layer	26
Gambar 2. 8 Dataset	27
Gambar 2. 9 Webcam.....	28
Gambar 2. 10 Layar Monitor.....	28
Gambar 3. 1 Diagram Alir Penelitian.....	30
Gambar 3. 2 Flowchart.....	32
Gambar 3. 3 Blok Diagram	35
Gambar 3. 4 Blok Diagram Pembuatan Model	36
Gambar 3. 5 Flowchart Pembuatan Model.....	37
Gambar 3. 6 Proses Pendeteksian Objek.....	38
Gambar 3. 7 Pemberian Nama Identitas Wajah	38
Gambar 3. 8 Proses Pengambilan Gambar Wajah.....	39
Gambar 3. 9 Proses Pengkondisian Dataset	39
Gambar 3. 10 Hasil Pengkondisian Dataset	40
Gambar 3. 11 Proses Pembagian Dataset	40
Gambar 3. 12 Proses Augmentasi Data.....	40
Gambar 3. 13 Hasil Proses Augmentasi Data	41
Gambar 3. 14 Pembuatan Model CNN.....	41
Gambar 3. 15 Proses Pelatihan Model	42
Gambar 3. 16 Proses Pembuatan Grafik.....	42
Gambar 3. 17 Hasil Kurva Accuracy dan Loss	43
Gambar 3. 18 Save Model.....	44
Gambar 3. 19 Pengujian model dengan dataset baru.....	44
Gambar 3. 20 Hasil Evaluasi Model.....	44
Gambar 3. 21 Peletakan Kamera Webcam.....	45
Gambar 3. 22 Posisi Webcam Dengan Dengan Wajah yang Akan Dideteksi.....	45
Gambar 3. 23 Proses Pembuatan Folder.....	46
Gambar 3. 24 Penyimpanan Hasil Identifikasi.....	47
Gambar 4. 1 Detected Faces	52
Gambar 4. 2 Tampilan Layar Monitor Identifikasi Wajah	52
Gambar 4. 3 Detected_faces.....	57
Gambar 4. 4 Tampilan Layar Monitor Identifikasi Wajah.....	57



DAFTAR TABEL

Tabel 2. 1 Penelitian Terdahulu Oleh Avrijsto et al.....	16
Tabel 2. 2 Pendahuluan Terdahulu Oleh Sutrarti et al	17
Tabel 2. 3 Penelitian Terdahulu Oleh Dufan et al.....	17
Tabel 2. 4 Penelitian Terdahulu Oleh.....	18
Tabel 2. 5 Spesifikasi Raspberry Pi 4B.....	19
Tabel 2. 6 Tabel Spesifikasi Kamera Webcam	28
Tabel 2. 7 Spesifikasi Layar Monitor.....	29
Tabel 3. 1 Spesifikasi Alat	34
Tabel 4. 1 Identitas Pengujian	48
Tabel 4. 2 Daftar Peralatan Pengujian 1.....	49
Tabel 4. 3 Data Hasil Percobaan Pada Pagi Hari	50
Tabel 4. 4 Identitas Pengujian	53
Tabel 4. 5 Daftar Alat Pengujian 1.....	54
Tabel 4. 6 Data Hasil Percobaan Pada Malam Hari	55

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

POLITEKNIK
NEGERI
JAKARTA



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

DAFTAR LAMPIRAN

Lampiran 1 DFTAR RIWAYAT HIDUP PENULIS	L-1
Lampiran 2 FOTO ALAT	64
Lampiran 3 LISTING PROGRAM	65





BAB I PENDAHULUAN

1.1 Latar Belakang

Teknologi pengenalan wajah telah mengalami perkembangan pesat, sistem pendeteksi wajah yang menggunakan kamera webcam sebagai citra, dimana citra adalah suatu representasi (gambaran), kemiripan atau imitasi dari suatu objek (Alwendi & Masriadi, 2021). Perkembangan teknologi pengenalan wajah atau dapat disebut juga sebagai *face recognition* dapat berdampak langsung di berbagai kegiatan salah satunya adalah sistem absensi yang digunakan sebagai bukti kehadiran siswa dalam mengikuti kegiatan di kampus. Absensi masih seringkali menggunakan metode manual dengan cara mencatat kehadiran menggunakan kertas, metode tersebut dirasa masih kurang efektif karena penggunaan kertas yang berpotensi hilang dan rusak. Salah satu kelebihan proses absensi dengan metode pengenalan wajah adalah ketika proses pendeteksian dilakukan secara realtime hasil dari pendeteksi identitas tersebut dapat disimpan ke dalam folder khusus agar bukti kehadiran tidak dapat dimanipulasi oleh siswa. Seringkali siswa melakukan manipulasi absensi, hal ini dapat dicegah dengan menambahkan perhitungan manusia untuk mendeteksi jumlah orang yang hadir secara fisik dalam ruangan.

Face recognition merupakan salah satu teknologi biometric, dimana kemampuan mengukur karakteristik fisik atau perilaku dapat digunakan untuk memverifikasi atau mengidentifikasi seseorang disebut dengan biometric (Setiawan dan Dewi Agushinta, 2020). Dalam pembuatan sistem *face recognition* dibutuhkan beberapa library yang dipakai salah satunya adalah *opencv* dan *tensorflow*. *Open Computer Vision (OpenCV)*, didefinisikan sebagai bidang studi yang berupaya mengembangkan Teknik untuk membantu computer melihat dan memahami onten gambar digital seperti foto dan video, tujuan dari *computer vision* adalah untuk memahami isi dari gambar digital (Wira Widjanarko et al., 2022). Salah satu penerapan *Open Computer Vision* adalah proses pengenalan wajah manusia atau *face recognition*. Dalam upaya pembangunan *face recognition* dapat menggunakan *Machine Learning* dengan salah satu framework yaitu *Tensorflow*. Dengan memanfaatkan pengolahan citra dan analisa data yang disediakan oleh *tensorflow* maka sistem yang digunakan dapat melakukan deteksi wajah dan identifikasi wajah secara realtime. *tensorflow* adalah sebuah antarmuka yang digunakan untuk ekspresi algoritma pembelajaran mesin dan

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

untuk eksekusi perintah dengan menggunakan informasi yang dimiliki tentang objek tersebut atau target yang akan dikenali, serta dapat membedakan objek satu dengan objek lainnya (Sapteka et al., 2022).

Tensorflow dapat memungkinkan pembangunan model deep learning untuk memahami dan mengidentifikasi wajah dengan tingkat akurasi yang tinggi karena menyediakan fleksibilitas dalam mendefinisikan dan menyesuaikan arsitektur model neural network. *Deep learning* merupakan metode learning yang memanfaatkan artificial neural network yang berlapis-lapis (multi layer), *Artificial Neural Network* dibuat mirip otak manusia, dimana neuron-neuron terkoneksi satu sama lain sehingga membentuk sebuah jaringan neuron yang sangat rumit (Adi Nugroho et al., 2020). Salah satu algoritma deep learning yang digunakan untuk membuat sistem identifikasi wajah yaitu adalah *Convolutional Neural Network* (CNN). *Convolutional Neural Network* (CNN) merupakan jenis arsitektur jaringan syaraf tiruan yang umumnya bekerja dengan cara melakukan pengenalan pola visual dan pengolahan citra. CNN digunakan untuk menganalisis gambar visual, mendeteksi dan mengenali objek pada *image* yang merupakan vektor berdimensi tinggi yang akan melibatkan banyak parameter untuk mencirikan jaringan (Adi Nugroho et al., 2020)

Berdasarkan permasalahan yang telah diuraikan, maka dalam tugas akhir ini akan dibuat alat yang dapat mengidentifikasi wajah menggunakan webcam dengan hasil deteksi yang akan disimpan ke dalam folder khusus ketika sedang melakukan pendeteksian secara realtime. Sistem identifikasi wajah ini akan dibuat menggunakan tensorflow sebagai framework untuk membangun model identifikasi wajah dan melakukan load model pada sebuah model *deep learning* yang dapat mendeteksi identitas dari tiap wajah apakah wajah tersebut sesuai identitasnya atau tidak dengan menggunakan metode CNN.

1.2 Perumusan Masalah

Berdasarkan latar belakang di atas diperoleh perumusan masalah sebagai berikut:

1. Bagaimana menerapkan tensorflow untuk mengambil setiap wajah yang digunakan sebagai dataset,
2. Bagaimana menerapkan tensorflow untuk membuat model deep learning dengan metode *Convolutional Neural Network* (CNN) pada sistem identifikasi wajah di dalam ruangan.



1.3 Tujuan

1. Menerapkan *framework* tensorflow sebagai pembuatan dataset yang memiliki nama beserta gambar wajah masing masing identitas,
2. Menerapkan *framework* tensorflow sebagai pembuatan model identifikasi wajah dengan metode *Convolutional Neural Network*(CNN).

1.4 Batasan Masalah

1. Pada sub judul hanya akan berfokus pada pembuatan dataset menggunakan tensorflow serta pengujian hasil model CNN
2. Tampilan kamera output hanya dapat dilihat melalui layar monitor
3. Menggunakan 3 objek sebagai dataset pada identifikasi wajah
4. Webcam diletakkan hanya pada satu sudut pandang saja tidak bergerak atau berubah tempat,
5. Pengujian dilakukan hanya pada saat lampu ruangan menyala,
6. Hanya mendeteksi wajah manusia tanpa disertai aksesoris yang menutupi wajah, seperti masker, topi, dan lain lain.

1.5 Luaran

1. Luaran Wajib
Sistem Identifikasi Wajah Seseorang Dalam Ruangan Menggunakan *Framework Tensorflow*
2. Laporan Akhir
 - Luaran Tambahan
 - Publikasi Media Sosial

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



BAB V KESIMPULAN

5.1. Kesimpulan

Adapun kesimpulan berdasarkan pengujian dan pembahasan yang sudah dilakukan yaitu :

1. Nilai akurasi pengujian jarak antara identitas sebenarnya dengan kamera webcam yang dilakukan pada pagi hari dengan intensitas cahaya 171 lux hingga 67 lux mendapatkan perhitungan tepat dengan hasil akurasi sebesar 92%. Jarak terbaik dalam percobaan ketika pagi hari didapatkan pada jarak 30cm dengan hasil 100%, sedangkan jarak 180cm mendapatkan hasil 40%
2. Nilai akurasi pada pengujian jarak antara identitas sebenarnya dengan kamera webcam yang dilakukan pada malam hari dengan intensitas cahaya 36 lux hingga 30 lux mendapatkan perhitungan tepat dengan hasil akurasi sebesar 84%. Jarak terbaik dalam percobaan ketika malam hari didapatkan pada jarak 30cm dengan hasil 100%, sedangkan jarak 180cm mendapatkan hasil 20%
3. Jarak antara identitas sebenarnya dengan webcam dapat mempengaruhi hasil kinerja sistem, sistem tidak dapat mendeteksi wajah dengan maksimal karena jarak antara wajah dengan kamera webcam terlalu jauh
4. Dari hasil 2 percobaan yang dilakukan pada pagi hari dan malam hari dapat disimpulkan bahwa proses pendeteksian memiliki hasil yang akurat jika mendapatkan cahaya yang cukup yaitu pada pengujian pagi hari dengan intensitas cahaya 67 lux hingga 171 lux, sedangkan pengujian yang dilakukan pada malam hari mendapatkan hasil akurasi yang kurang bagus dibanding dengan hasil akurasi pada pagi hari diakibatkan intensitas cahaya yang kurang sehingga model tidak dapat memprediksi identitas wajah dengan akurat
5. Kinerja model identifikasi wajah yang dibuat menggunakan metode Convolutuona Neural Network (CNN) sudah cukup bagus jika mendapatkan cahaya yang baik dan jarak yang tidak terlalu jauh pada webcam

5.2. Saran

Berdasarkan hasil penelitian dan pengujian yang telah dilakukan, terdapat beberapa saran untuk membuat sistem identifikasi wajah

1. Diharapkan penelitian selanjutnya untuk menggunakan kamera webcam yang lebih bagus agar mendapatkan input yang baik sehingga proses identifikasi lebih baik,
2. Diharapkan penelitian selanjutnya menggunakan mikrokomputer dengan processor yang lebih besar agar pada saat proses identifikasi wajah tidak mengalami delay yang terlalu lama.



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta





DAFTAR PUSTAKA

37-Article Text-150-1-10-20191008. (n.d.).

Adi Nugroho, P., Fenriana, I., & Arijanto, R. (2020). Implementasi Deep Learning Menggunakan Convolutional Neural Network (CNN) Pada Ekspresi Manusia. *Jurnal Algor*, 2(1). <https://jurnal.buddhidharma.ac.id/index.php/algor/index>

Alwendi, & Masriadi. (2021). Aplikasi Pengenalan Wajah Manusia Pada Citra Menggunakan Metode Fisherface (Vol. 11, Issue 1).

Amandri Achyar, A., Muhammad Olow, A., Rizky Perdana, M., Sundawijaya, A., & Dhiyaanisafa Goenawan, A. (2022). *Identifikasi Ras Wajah dengan Menggunakan Metode Deep Learning Model Keras* (Vol. 1, Issue 1).

Annur, I. F., Umami, J., Annafii, Moch. N., Trisnaningrum, N., & Putra, O. V. (2023). Klasifikasi Tingkat Keparahan Penyakit Leafblast Tanaman Padi Menggunakan MobileNetv2. *Fountain of Informatics Journal*, 8(1), 7–14. <https://doi.org/10.21111/fij.v8i1.9419>

Arsal, M., Agus Wardijono, B., & Anggraini, D. (2020). Face Recognition Untuk Akses Pegawai Bank Menggunakan Deep Learning Dengan Metode CNN. *Jurnal Nasional Teknologi Dan Sistem Informasi*, 6(1), 55–63. <https://doi.org/10.25077/teknosi.v6i1.2020.55-63>

Darmawan Sidik, A., Ansawarman, A., Kunci, K., Kendaraan Bermotor, J., Regresi, M., & Jalan, F. (2022). Prediksi Jumlah Kendaraan Bermotor Menggunakan Machine Learning. *Formosa Journal of Multidisciplinary Research (FJMR)*, 1(3), 559–568. <https://doi.org/10.55927>

Eka, A., #1, P., Farid, M., #2, N., Riandaru, V., #3, P., Kalirungkut, J. R., Rungkut, K., Rungkut, K., Sby, K., & Timur, J. (2023). *JEPIN (Jurnal Edukasi dan Penelitian Informatika) Klasifikasi Jenis Rempah Menggunakan Convolutional Neural Network dan Transfer Learning*.

Faktor-Faktor yang Mempengaruhi Sistem Pengenalan Wajah Menggunakan Metode Eigenface pada Perangkat Mobile Berbasis Android Derisma, *jurnalpcracid*.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

(2016). *Jurnal Politeknik Caltex Riau*. In *Jurnal Komputer Terapan* (Vol. 2, Issue 2). <http://jurnal.pcr.ac.id>

Haris, M., Pustaka, T., Diponegoro, M. H., Kusumawardani, S., & Hidayah, I. (2021). Tinjauan Pustaka Sistematis: Implementasi Metode Deep Learning pada Prediksi Kinerja Murid (Implementation of Deep Learning Methods in Predicting Student Performance: A Systematic Literature Review). In *Jurnal Nasional Teknik Elektro dan Teknologi Informasi* / (Vol. 10, Issue 2).

Juli Irzal Ismail, S., & Bayu Satrya, G. (n.d.). Perancangan Sistem Keamanan Rumah Menggunakan Deteksi Wajah Berbasis Machine Learning Menggunakan Tensorflow *Designing a Home Security System Using Machine Learning Based Face Detection Using Tensorflow*.

Khunafa Qudsi, N., Asmara, R. A., Syulistyo, A. R., Studi, P., Informatika, T., Informasi, J. T., & Malang, P. N. (n.d.). *Identifikasi Citra Tulisan Tangan Digital Menggunakan Convolutional Neural Network (CNN)*.

Malik, M. (2022). *Deteksi Suhu Tubuh dan Masker Wajah dengan MLX90614, Opencv, Keras/Tensorflow, dan Deep Learning*. 6(1).

Miftahuddin, Y., & Zaelani, F. (2022). Perbandingan Metode Efficientnet-B3 dan Mobilenet-V2 Untuk Identifikasi Jenis Buah-buahan Menggunakan Fitur Daun. In *Fiqry Zaelani S Jurnal Ilmiah Teknologi Informasi Terapan* (Vol. 9, Issue 1).

Putra, H. E., Jamil, M., & Lutfi, S. (2019). Smart Akuarium Berbasis IoT Menggunakan Raspberry Pi3. *Jurnal Informatika Dan Komputer* p-ISSN, 2(2), 2355–7699. <https://doi.org/10.33387/jiko>

Sapteka, A. A. N. G., Dharma, P. A. S., Widyatmika, K. A., Suparta, I. N., Yasa, I. M. S., & Sapteka, A. A. N. G. (2022). Pendeteksi Penggunaan Masker Wajah dengan ESP32Cam Menggunakan OpenCV dan Tensorflow. *Majalah Ilmiah Teknologi Elektro*, 21(2), 155. <https://doi.org/10.24843/mite.2022.v21i02.p01>

Satwikayana, S., Wibowo, S. A., & Vendyansyah, N. (2021). Sistem Presensi Mahasiswa Otomatis Pada Zoom Meeting Menggunakan Face Recognition Dengan Metode Convolutional Neural Network Berbasis Web. In *Jurnal Mahasiswa Teknik Informatika* (Vol. 5, Issue 2).



Setiawan dan Dewi Agushinta, F. R. (2020). Sistem Pengenalan Wajah Dengan Metode Local Binary Ptern Histogram Pada Firebase Berbasis OpenCV. *Seminar Nasional Teknologi Informasi Dan Komunikasi STI&K (SeNTIK)*, 4(1), 16424.

Sirait, F. (n.d.). *Sistem Monitoring Keamanan Gedung berbasis Rasberry Pi*.

Susim, T., Darujati, C., & Artikel, I. (2021). Pengolahan Citra Untuk Pengenalan Wajah (Face recognition) Menggunakan OpenCV. *Jurnal Syntax Admiration*, 2(3).

Wira Widjanarko, K., Aditya Herlambang, K., Abdul Karim, M., & Artikel Abstrak, I. (2022). Faster Video Processing Menggunakan Teknik Parallel Processing Dengan Library OpenCV. In *Sains dan Teknologi* (Vol. 1, Issue 1). <https://journal.proletargroup.org/index.php/JKST>



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

LAMPIRAN

Lampiran 1

DFTAR RIWAYAT HIDUP PENULIS

DIFNA YASMIKA

Anak tunggal. Lahir di Tegal, 4 September 2001. Lulus dari SDIT Attaqwa Pusat 2013, lulus Pendidikan menengah pertama di SMP-IT Attaqwa tahun 2016, lulus Pendidikan menengah atas di SMAN 1 Babelan tahun 2019. Gelar Diploma Tiga (D3) diperoleh pada tahun 2022 dari jurusan Teknik Elektro, Program Studi Elektronika Industri, Politeknik Negeri Jakarta. Penulis melanjutkan Pendidikan ke jenjang Sarjana Terapan (D4) di Politeknik Negeri Jakarta jurusan Teknik Elektro program studi Instrumentasi dan Kontrol Industri pada tahun 2022.



**POLITEKNIK
NEGERI
JAKARTA**



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Lampiran 2

FOTO ALAT



Gambar L. 1 Foto tampak samping kiri



Gambar L. 2 foto tampak samping kanan



Gambar L. 3 Tampilan Layar Monitor

© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta





- Hak Cipta :**
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
 2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Lampiran 3

LISTING PROGRAM

- **Program Pembuatan Model CNN**

```

import os

import numpy as np

import cv2

import pandas as pd

import matplotlib.pyplot as plt

from matplotlib.pyplot import imshow

from sklearn.preprocessing import LabelEncoder

from sklearn.model_selection import train_test_split

from sklearn.metrics import classification_report

from sklearn.metrics import confusion_matrix

import tensorflow as tf

from tensorflow import keras

from tensorflow.keras import backend as K

from tensorflow.keras.models import Sequential, Model

from tensorflow.keras.layers import Dense, Activation, Input

from tensorflow.keras.layers import Conv2D, MaxPool2D, Flatten

from tensorflow.keras.utils import to_categorical

from tensorflow.keras.preprocessing.image import ImageDataGenerator, load_img, img_to_array

sdir=r'/content/drive/MyDrive/TA D4/datasetN'

classlist=os.listdir(sdir)

label_file_path = "/content/drive/MyDrive/TA D4/program keseluruhan/labels2.txt"

with open(label_file_path, 'w') as file:

    for label in classlist:

        file.write(label + '\n')

filepaths=[]

labels=[]

for klass in classlist:

    classpath=os.path.join(sdir, klass)

    if os.path.isdir(classpath):
  
```



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
flist=os.listdir(classpath)

for f in flist:

    fpath=os.path.join(classpath,f)

    filepaths.append(fpath)

    labels.append(klass)

Fseries= pd.Series(filepaths, name='filepaths')

Lseries=pd.Series(labels, name='labels')

df=pd.concat([Fseries, Lseries], axis=1)

train_split=0.8

test_split=0.1

dummy_split=test_split/(1-train_split)

train_df, dummy_df=train_test_split(df, train_size=train_split, shuffle=True, random_state=123)

test_df, valid_df=train_test_split(dummy_df, train_size=dummy_split, shuffle=True,
random_state=123)

print ('train_df length: ', len(train_df), ' test_df length: ', len(test_df), ' valid_df length: ',
len(valid_df))

height=224

width=224

channels=3

batch_size=64

img_shape=(height, width, channels)

img_size=(height, width)

length=len(test_df)

test_batch_size=sorted([int(length/n) for n in range(1,length+1) if length % n ==0 and
length/n<=80],reverse=True)[0]

test_steps=int(length/test_batch_size)

gen=ImageDataGenerator(rescale=1./255, rotation_range = 20, brightness_range =(0.8,0.99))

train_gen=gen.flow_from_dataframe( train_df, x_col='filepaths', y_col='labels',
target_size=img_size, class_mode='categorical', color_mode='rgb', shuffle=True,
batch_size=batch_size)

validgen=ImageDataGenerator(rescale=1./255)

valid_gen=validgen.flow_from_dataframe( valid_df, x_col='filepaths', y_col='labels',
target_size=img_size, class_mode='categorical', color_mode='rgb', shuffle=True,
batch_size=batch_size)

testgen=ImageDataGenerator(rescale=1./255)
```




Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

test_gen=testgen.flow_from_dataframe( test_df, x_col='filepaths', y_col='labels',
target_size=img_size, class_mode='categorical', color_mode='rgb', shuffle=False,
batch_size=test_batch_size)

classes=list(train_gen.class_indices.keys())

class_count=len(classes)

base_model=tf.keras.applications.MobileNetV2(include_top=False, weights="imagenet",,
input_tensor=Input(shape=(224,224,3)))

model_name='model1'

model = tf.keras.Sequential([

    base_model,

    tf.keras.layers.Conv2D(filters=32, kernel_size=3, activation='relu', padding='same', strides=1),

    tf.keras.layers.MaxPooling2D(pool_size=2, strides=2),

    tf.keras.layers.Dropout(rate=0.25),

    tf.keras.layers.Flatten(),

    tf.keras.layers.Dense(128, activation='relu'),

    tf.keras.layers.Dropout(rate=0.5),

    tf.keras.layers.Dense(3, activation='softmax')])

model.compile(optimizer=tf.keras.optimizers.Adam(learning_rate=0.00003),
loss='categorical_crossentropy', metrics='accuracy')

model_name = 'Model1'

epochs = 100

batch_size = 150

history=model.fit(x=train_gen, epochs=epochs, batch_size=batch_size,
validation_data=valid_gen)

def tr_plot(tr_data, start_epoch):

    tacc=tr_data.history['accuracy']

    tloss=tr_data.history['loss']

    vacc=tr_data.history['val_accuracy']

    vloss=tr_data.history['val_loss']

    Epoch_count=len(tacc)+ start_epoch

    Epochs=[]

    for i in range (start_epoch ,Epoch_count):

        Epochs.append(i+1)

    index_loss=np.argmin(vloss)# this is the epoch with the lowest validation loss

    val_lowest=vloss[index_loss]

```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

index_acc=np.argmax(vacc)
acc_highest=vacc[index_acc]

plt.style.use('fivethirtyeight')

sc_label='best epoch= '+ str(index_loss+1 +start_epoch)
vc_label='best epoch= '+ str(index_acc + 1+ start_epoch)

fig,axes=plt.subplots(nrows=1, ncols=2, figsize=(20,8))
axes[0].plot(Epochs,tloss, 'r', label='Training loss')
axes[0].plot(Epochs,vloss,'g',label='Validation loss' )
axes[0].set_title('Training and Validation Loss')
axes[0].set_xlabel('Epochs')
axes[0].set_ylabel('Loss')
axes[0].legend()
axes[1].plot (Epochs,tacc,'r',label= 'Training Accuracy')
axes[1].plot (Epochs,vacc,'g',label= 'Validation Accuracy')
axes[1].set_title('Training and Validation Accuracy')
axes[1].set_xlabel('Epochs')
axes[1].set_ylabel('Accuracy')
axes[1].legend()

plt.tight_layout
plt.show()

tr_plot(history,0)

save_dir=r'/content/drive/MyDrive/TA D4'
subject='modelcnn4'

acc=model.evaluate( test_gen, batch_size=test_batch_size, verbose=1, steps=test_steps,
return_dict=False)[1]*100

msg=f'accuracy on the test set is {acc:5.2f} %'

print_in_color(msg, (0,255,0),(55,65,80))

save_id=str (model_name + '-' + subject +'-' + str(acc)[:str(acc).rfind('.')+3] + '.h5')

save_loc=os.path.join(save_dir, save_id)

model.save(save_loc)

import seaborn as sns

def print_info( test_gen, preds, print_code, save_dir, subject ):

    class_dict=test_gen.class_indices
  
```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

labels= test_gen.labels
file_names= test_gen filenames
error_list=[]
true_class=[]
pred_class=[]
prob_list=[]
new_dict={ }
error_indices=[]
y_pred=[]
for key,value in class_dict.items():
    new_dict[value]=key      # dictionary {integer of class number: string of class name}
# store new_dict as a text file in the save_dir
classes=list(new_dict.values()) # list of string of class names
dict_as_text=str(new_dict)
dict_name= subject + '-' +str(len(classes)) +'.txt'
dict_path=os.path.join(save_dir,dict_name)
with open(dict_path, 'w') as x_file:
    x_file.write(dict_as_text)
errors=0
for i, p in enumerate(preds):
    pred_index=np.argmax(p)
    true_index=labels[i] # labels are integer values
    if pred_index != true_index: # a misclassification has occurred
        error_list.append(file_names[i])
        true_class.append(new_dict[true_index])
        pred_class.append(new_dict[pred_index])
        prob_list.append(p[pred_index])
        error_indices.append(true_index)
        errors=errors + 1
    y_pred.append(pred_index)
if print_code !=0:
    if errors>0:

```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

if print_code>errors:
    r=errors
else:
    r=print_code
    msg='{0:^28s}{1:^28s}{2:^28s}{3:^16s}'.format('Filename', 'Predicted Class', 'True
Class', 'Probability')
    print_in_color(msg, (0,255,0),(55,65,80))
for i in range(r):
    split1=os.path.split(error_list[i])
    split2=os.path.split(split1[0])
    fname=split2[1] + '/' + split1[1]
    msg='{0:^28s}{1:^28s}{2:^28s}{3:4s}{4:^6.4f}'.format(fname,
pred_class[i],true_class[i],',', prob_list[i])
    print_in_color(msg, (255,255,255), (55,65,60))
else:
    msg='With accuracy of 100 % there are no errors to print'
    print_in_color(msg, (0,255,0),(55,65,80))
if errors>0:
    plot_bar=[]
    plot_class=[]
    for key, value in new_dict.items():
        count=error_indices.count(key)
        if count!=0:
            plot_bar.append(count) # list containg how many times a class c had an error
            plot_class.append(value) # stores the class
    fig=plt.figure()
    fig.set_figheight(len(plot_class)/3)
    fig.set_figwidth(10)
    plt.style.use('fivethirtyeight')
    for i in range(0, len(plot_class)):
        c=plot_class[i]
        x=plot_bar[i]

```



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

plt.barh(c, x, )

plt.title( ' Errors by Class on Test Set')

y_true= np.array(labels)
y_pred=np.array(y_pred)
if len(classes)<= 30:
    # create a confusion matrix
    cm = confusion_matrix(y_true, y_pred )
    length=len(classes)
    if length<8:
        fig_width=8
        fig_height=8
    else:
        fig_width= int(length * .5)
        fig_height= int(length * .5)
    plt.figure(figsize=(fig_width, fig_height))
    sns.heatmap(cm, annot=True, vmin=0, fmt='g', cmap='Blues', cbar=False)
    plt.xticks(np.arange(length)+.5, classes, rotation= 90)
    plt.yticks(np.arange(length)+.5, classes, rotation=0)
    plt.xlabel("Predicted")
    plt.ylabel("Actual")
    plt.title("Confusion Matrix")
    plt.show()
    clr = classification_report(y_true, y_pred, target_names=classes)
    print("Classification Report:\n-----\n", clr)

print_code=0

preds=model.predict(test_gen)

print_info( test_gen, preds, print_code, save_dir, subject )

```

Program Lebur

```

try:
    log = open('log.txt','w')
except:

```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

print( "Tidak dapat membuka file log")

cnt_up  = 0

cnt_down = 0

cap = cv2.VideoCapture(2)

face_cascade = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')

np.set_printoptions(suppress=True)

output_folder = "detected_faces"

if not os.path.exists(output_folder):

    os.makedirs(output_folder)

model = load_model("model1.h5")

labels = open("modelcnn-3.txt", "r").readlines()

print("[INFO] Model berhasil dimuat...")

camera = cv2.VideoCapture(0)

font = cv2.FONT_HERSHEY_COMPLEX

camera.set(cv2.CAP_PROP_FRAME_WIDTH, 480)

camera.set(cv2.CAP_PROP_FRAME_HEIGHT, 320)

for i in range(19):

    print( i, cap.get(i))

h = 480

w = 640

frameArea = h*w

areaTH = frameArea/6

print( 'Area Threshold', areaTH)

line_up = int(2*(h/5))

line_down  = int(3*(h/5))

up_limit =  int(1*(h/5))

down_limit = int(4*(h/5))

print( "Red line y:",str(line_down))

print( "Blue line y:", str(line_up))

line_down_color = (255,0,0)

line_up_color = (0,0,255)

pt1 = [0, line_down];

```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

pt2 = [w, line_down];
pts_L1 = np.array([pt1,pt2], np.int32)
pts_L1 = pts_L1.reshape((-1,1,2))
pt3 = [0, line_up];
pt4 = [w, line_up];
pts_L2 = np.array([pt3,pt4], np.int32)
pts_L2 = pts_L2.reshape((-1,1,2))
pt5 = [0, up_limit];
pt6 = [w, up_limit];
pts_L3 = np.array([pt5,pt6], np.int32)
pts_L3 = pts_L3.reshape((-1,1,2))
pt7 = [0, down_limit];
pt8 = [w, down_limit];
pts_L4 = np.array([pt7,pt8], np.int32)
pts_L4 = pts_L4.reshape((-1,1,2))
fgbg = cv2.createBackgroundSubtractorMOG2(detectShadows = True)
kernelOp = np.ones((3,3),np.uint8)
kernelOp2 = np.ones((5,5),np.uint8)
kernelCl = np.ones((11,11),np.uint8)
font = cv2.FONT_HERSHEY_SIMPLEX
persons = []
max_p_age = 5
pid = 1
while cap.isOpened():
    ret, img = camera.read()
    ret1, frame = cap.read()
    faces = face_cascade.detectMultiScale(gray, 1.1, 5)
    for (x, y, w, h) in faces:
        face_img = cv2.resize(img[y:y + h, x:x + w], (224, 224), interpolation=cv2.INTER_AREA)
        face_img = np.asarray(face_img, dtype=np.float32).reshape(1, 224, 224, 3)
        face_img = (face_img / 127.5) - 1
        result = model.predict(face_img)

```



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

idx = np.argmax(result)
print(idx)
class_name = labels[idx].strip()
confidence = result.max(axis=1) * 100
if confidence > 90:
    label_text = "%s (%.2f %%) " % (class_name[2:], confidence)
    cv2.rectangle(img, (x, y), (x + w, y + h), (0, 255, 0), 2)
    cv2.rectangle(img, (x, y - 40), (x + w, y), (0, 255, 0), -2)
    cv2.putText(img, label_text, (x, y - 10), font, 0.75, (255, 255, 255), 1, cv2.LINE_AA)
    timestamp = time.strftime("%Y%m%d_%H.%M%p")
    safe_label = label_text.replace(' ', '').replace(':', '').replace('.', '').replace('-', '')
    cv2.imwrite(os.path.join(output_folder, f"{label_text}_{timestamp}.jpg"), img)
else:
    label_text = "Unknown"
    cv2.rectangle(img, (x, y), (x + w, y + h), (0, 255, 0), 2)
    cv2.rectangle(img, (x, y - 40), (x + w, y), (0, 255, 0), -2)
    cv2.putText(img, label_text, (x, y - 10), font, 0.75, (255, 255, 255), 1, cv2.LINE_AA)
    timestamp = time.strftime("%Y%m%d_%H.%M%p")
    safe_label = label_text.replace(' ', '').replace(':', '').replace('.', '').replace('-', '')
    cv2.imwrite(os.path.join(output_folders, f"{label_text}_{timestamp}.jpg"), img)
for i in persons:
    i.age_one()
fgmask = fgbg.apply(frame)
fgmask2 = fgbg.apply(frame)
try:
    ret,imBin= cv2.threshold(fgmask,130,255,cv2.THRESH_BINARY)
    ret,imBin2 = cv2.threshold(fgmask2,130,255,cv2.THRESH_BINARY)
    mask = cv2.morphologyEx(imBin, cv2.MORPH_OPEN, kernelOp)
    mask2 = cv2.morphologyEx(imBin2, cv2.MORPH_OPEN, kernelOp)
    mask = cv2.morphologyEx(mask , cv2.MORPH_CLOSE, kernelCl)
    mask2 = cv2.morphologyEx(mask2, cv2.MORPH_CLOSE, kernelCl)
except:

```




Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

print('EOF')

print( 'Masuk:',cnt_up)

print ('Keluar:',cnt_down)

break

contours0,hierarchy=
cv2.findContours(mask2,cv2.RETR_EXTERNAL,cv2.CHAIN_APPROX_SIMPLE)

for cnt in contours0:

    area = cv2.contourArea(cnt)

    if area > areaTH:

        M = cv2.moments(cnt)
        cx = int(M['m10']/M['m00'])
        cy = int(M['m01']/M['m00'])

        x,y,w,h = cv2.boundingRect(cnt)

        new = True

        if cy in range(up_limit,down_limit):

            for i in persons:

                if abs(x-i.getX()) <= w and abs(y-i.getY()) <= h:

                    #objek tersebut dekat dengan objek yang telah terdeteksi sebelumnya

                    new = False

                    i.updateCoords(cx,cy) #memperbarui koordinat dalam objek dan mengatur ulang

                    if i.going_UP(line_down,line_up) == True:

                        cnt_up += 1;

                        print( "ID:",i.getId(),'crossed going up at',time.strftime("%c"))

                        log.write("ID: "+str(i.getId())+' crossed going up at ' + time.strftime("%c") + "\n")

                    elif i.going_DOWN(line_down,line_up) == True:

                        cnt_down += 1;

                        print( "ID:",i.getId(),'crossed going down at',time.strftime("%c"))

                        log.write("ID: " + str(i.getId()) + ' crossed going down at ' + time.strftime("%c")

                        + '\n')

                    break

            if i.getState() == '1':

                if i.getDir() == 'down' and i.getY() > down_limit:

                    i.setDone()

```



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

elif i.getDir() == 'up' and i.getY() < up_limit:

    i.setDone()

if i.timedOut():

    index = persons.index(i)

    persons.pop(index)

    del i #hapus i

if new == True:

    p = Person.MyPerson(pid,cx,cy, max_p_age)

    persons.append(p)

    pid += 1

cv2.circle(frame,(cx,cy), 5, (0,0,255), -1)

img = cv2.rectangle(frame,(x,y),(x+w,y+h),(0,255,0),2)

#cv2.drawContours(frame, cnt, -1, (0,255,0), 3)

for i in persons:

    cv2.putText(frame, str(i.getId()),(i.getX(),i.getY()),font,0.3,i.getRGB(),1,cv2.LINE_AA)

str_up = 'Masuk: '+ str(cnt_up)

str_down = 'Keluar: '+ str(cnt_down)

frame = cv2.polylines(frame,[pts_L1],False,line_down_color,thickness=2)

frame = cv2.polylines(frame,[pts_L2],False,line_up_color,thickness=2)

frame = cv2.polylines(frame,[pts_L3],False,(255,255,255),thickness=1)

frame = cv2.polylines(frame,[pts_L4],False,(255,255,255),thickness=1)

cv2.putText(frame, str_up ,(10,40),font,0.5,(255,255,255),2,cv2.LINE_AA)

cv2.putText(frame, str_up ,(10,40),font,0.5,(0,0,255),1,cv2.LINE_AA)

cv2.putText(frame, str_down ,(10,90),font,0.5,(255,255,255),2,cv2.LINE_AA)

cv2.putText(frame, str_down ,(10,90),font,0.5,(255,0,0),1,cv2.LINE_AA)

cv2.imshow("Hasil", img)

cv2.imshow('Frame',frame)

k = cv2.waitKey(30) & 0xff

if k == 27:

    break

log.flush()

log.close()

```

```
camera.release()  
cap.release()  
cv2.destroyAllWindows()
```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

