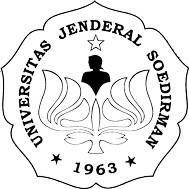
# 

PROPOSAL TUGAS AKHIR

KLASIFIKASI CT SCAN COVID-19 BERDASARKAN CITRA DENGAN MENGGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK (CNN)

Disusun untuk memenuhi kebutuhan akan templat berkas laporan  
di Jurusan Teknik Elektro Universitas Jenderal Soedirman



Disusun oleh:

Rokhi Iman Sarofi

H1A017065

**KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI**

**UNIVERSITAS JENDERAL SOEDIRMAN**

**FAKULTAS TEKNIK**

**JURUSAN/PROGRAM STUDI TEKNIK ELEKTRO**

**PURBALINGGA**

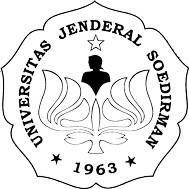
**2020**

# HALAMAN JUDUL

LAPORAN TUGAS AKHIR

KLASIFIKASI CT SCAN COVID-19 BERDASARKAN CITRA DENGAN MENGGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK (CNN)

Disusun untuk memenuhi kebutuhan akan templat berkas laporan  
di Jurusan Teknik Elektro Universitas Jenderal Soedirman



Disusun oleh:

Rokhi Iman Sarofi

H1A017065

**KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI**

**UNIVERSITAS JENDERAL SOEDIRMAN**

**FAKULTAS TEKNIK**

**JURUSAN/PROGRAM STUDI TEKNIK ELEKTRO**

**PURBALINGGA**

**2020**

# HALAMAN PENGESAHAN

Tugas Akhir dengan Judul:

KLASIFIKASI CT SCAN COVID-19 BERDASARKAN CITRA DENGAN MENGGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK (CNN)



Disusun oleh:

Rokhi Iman Sarofi  
H1A017065

Diajukan untuk memenuhi salah satu persyaratan  
memperoleh gelar Sarjana Teknik pada  
Jurusan/Program Studi Teknik Elektro  
Fakultas Teknik  
Universitas Jenderal Soedirman

Diterima dan disetujui  
Pada Tanggal : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |  |
| --- | --- | --- |
| Pembimbing I  Imron Rosyadi, S.T., M.Sc.  NIP : 197909242003121003 |  | Pembimbing II/Lapangan  Muhammad Syaiful Aliim, S.T., M.T.  (NIP : 199009052019031021) |
| Mengetahui:  Dekan Fakultas Teknik  Dr. Eng. Suroso, S.T., M.Eng.  NIP. 197812242001121002 | | |

# HALAMAN PERNYATAAN

Dengan ini saya menyatakan bahwa dalam Laporan Tugas Akhir[[1]](#footnote-1) dengan judul ***“KLASIFIKASI CT SCAN COVID-19 BERDASARKAN CITRA DENGAN MENGGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK (CNN)”*** ini tidak terdapat karya yang pernah diajukan untuk memperoleh gelar kesarjanaaan di suatu Perguruan Tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

|  |  |  |
| --- | --- | --- |
|  |  | Purbalingga, 5 Februari 2021  [materai sesuai ketentuan uu]  Ttd.  Rokhi Iman Sarofi  NIM. H1A017065 |

# HALAMAN MOTTO DAN PERSEMBAHAN

**MOTTO**

*“When there is will, there is way.”.*

**PERSEMBAHAN**

Laporan Tugas Akhir ini dapat diselesaikan atas dorongan, saran, serta  
bantuan pemikiran berbagai pihak. Pada kesempatan ini disampaikan ucapan  
terima kasih kepada :

1. Allah SWT, yang telah melimpahkan rahmat dan petunjuk selama pelaksanaan Tugas Akhir.
2. Kedua Orang Tua dan Saudara penulis atas dukungan baik moril maupun  
   materil selama pelaksanaan Tugas Akhir.
3. Ibu Farida Asriani, S.Si., M.T. selaku Ketua Jurusan Teknik Elektro
4. Bapak Imron Rosyadi, selaku pembimbing I.
5. BapakMuhammad Syaiful Aliim selaku pembimbing II
6. Semua pihak yang telah membantu baik secara langsung maupun tidak langsung dalam pelaksanaan Tugas Akhir.

# RINGKASAN

KLASIFIKASI CT SCAN COVID-19 BERDASARKAN CITRA DENGAN MENGGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK (CNN)

Rokhi Iman Sarofi

Coronavirus Disease 2019 (COVID-19) merupakan penyakit pernafasan akut yang disebabkan oleh virus corona jenis baru dan dapat menular dengan cepat melalui droplet. Penyakit ini memiliki gejala umum antara lain gejala gangguan pernapasan akut seperti demam, batuk, dan sesak napas. COVID-19 pertama ditemukan pada Desember 2019, di Wuhan, China.

Hingga saat ini proses untuk mendiagnosa dan konfirmasi *COVID-19* bergantung pada *real-time reverse-transcription–polymerase-chain-reaction* (RT-PCR) yang mendeteksi keberadaan *SARS-CoV-2*.[1] Selain konfirmasi hasil *RT-PCR*, elemen diagnostik utama lainnya yang dapat memfasilitasi identifikasi *COVID-19* adalah citra tomografi komputasi dada *(CT-Scan).*

Berdasarkan hal ini penulis ingin merancang sistem klasifikasi *CT Scan* untuk *COVID-19* berdasarkan citra dengan menggunakan metode convolutional neural network. Ada beberapa arsitektur CNN yang digunakan seperti *VGG16, MobileNet*, dan *ResNet*, kemudian penulis akan membandingkan hasil dari masing-masing arsitektur yang digunakan.

Kata kunci : *COVID-19*, *real-time reverse-transcription–polymerase-chain-reaction,CT Scan,* *CNN*

# *SUMMARY*

***CLASSIFICATION OF COVID-19 CT SCAN BASED ON IMAGE USING CONVOLUTIONAL NEURAL NETWORK (CNN)***

Rokhi Iman Sarofi

*Coronavirus Disease 2019 (COVID-19) is an acute respiratory disease caused by a new type of coronavirus and can be transmitted quickly through droplets. This disease has common symptoms including acute respiratory symptoms such as fever, cough and shortness of breath. The first COVID-19 was discovered in December 2019, in Wuhan, China.*

*Until now the process to diagnose and confirm COVID-19 has relied on real-time reverse-transcription-polymerase-chain-reaction (RT-PCR) that detects the presence of SARS-CoV-2. [2] Apart from confirming RT-PCR results, another key diagnostic element that can facilitate identification of COVID-19 is a computed tomography image of the chest (CT-Scan).*

*Based on this, the writer wants to design a CT Scan classification system for COVID-19 based on images using the convolutional neural network method. There are several CNN architectures used such as VGG16, MobileNet, and ResNet, then the author will compare the results of each architecture used.*

*Keywords : COVID-19*, *real-time reverse-transcription–polymerase-chain-reaction,CT Scan,* *CNN*

# PRAKATA

Puji syukur kehadirat Allah SWT karena atas segala berkah, rahmat, dan  
hidayah-Nya penulis dapat menyelesaikan laporan Tugas Akhir dengan judul **“Klasifikasi CT Scan COVID-19 Berdasarkan Citra dengan Menggunakan Metode Convolutional Neural Network (CNN)”** dengan baik dan tepat waktu.

Laporan Tugas Akhir ini disusun sebagai salah satu syarat mata kuliah  
Tugas Akhir pada program studi Teknik Elektro - Universitas Jenderal Soedirman  
pada tahun 2020 yang sedang penulis jalani. Laporan Tugas Akhir ini dapat  
diselesaikan atas dorongan, saran, serta bantuan pemikiran berbagai pihak. Pada  
kesempatan ini disampaikan ucapan terima kasih kepada : Kedua orang tua, Ibu  
Farida Asriani, S.Si., M.T. selaku Ketua Jurusan Teknik Elektro Unsoed dan dosen  
pembimbing Tugas Akhir, dan segenap rekan kerja selama Tugas Akhir yang telah sabar membimbing dan banyak memberikan ilmu, sahabat-sahabat  
yang selalu memberikan dukungan, semangat, motivasi, serta doa dan semua  
pihak yang telah membantu dalam penelitian.

Akhir kata, Penulis menyadari bahwa masih banyak terdapat kekurangan  
dalam laporan Tugas Akhir ini, maka kritik dan saran yang membangun sangat  
diharapkan dari berbagai pihak. Akhir kata penulis berharap semoga Laporan ini  
dapat bermanfaat bagi semua yang membutuhkannya, terutama bagi yang akan  
menyusun Laporan Tugas Akhir terkait pada periode selanjutnya.

Purbalingga, 5 Februari 2021

Rokhi Iman Sarofi

# DAFTAR ISI

[HALAMAN JUDUL i](#_Toc56525500)

[HALAMAN PENGESAHAN ii](#_Toc56525501)

[HALAMAN PERNYATAAN iii](#_Toc56525502)

[HALAMAN MOTTO DAN PERSEMBAHAN iv](#_Toc56525503)

[RINGKASAN v](#_Toc56525504)

[*SUMMARY* vi](#_Toc56525505)

[PRAKATA vii](#_Toc56525506)

[DAFTAR ISI viii](#_Toc56525507)

[DAFTAR TABEL x](#_Toc56525508)

[DAFTAR GAMBAR xi](#_Toc56525509)

[DAFTAR LAMPIRAN xii](#_Toc56525510)

[DAFTAR ISTILAH DAN SINGKATAN xiii](#_Toc56525511)

[DAFTAR SIMBOL xiv](#_Toc56525512)

[BAB 1 PENDAHULUAN 15](#_Toc56525513)

[1.1 Latar Belakang 15](#_Toc56525514)

[1.2 Rumusan Masalah 16](#_Toc56525515)

[1.3 Batasan Masalah 17](#_Toc56525516)

[1.4 Tujuan Penelitian 17](#_Toc56525517)

[1.5 Manfaat Penelitian 18](#_Toc56525518)

[1.6 Sistematika Penulisan 18](#_Toc56525519)

[BAB 2 TINJAUAN PUSTAKA 20](#_Toc56525520)

[2.1 Penelitian Terdahulu 20](#_Toc56525521)

[2.2 Pengolahan Citra 21](#_Toc56525522)

[*2.3 Deep Learning* 21](#_Toc56525523)

[*2.4 Convolutional Neural Network* 23](#_Toc56525524)

[*2.4.1 Input Layer* 24](#_Toc56525525)

[*2.4.2 Convolutional Layer* 24](#_Toc56525526)

[*2.4.3 Activation Layer* 25](#_Toc56525527)

[*2.4.4 Pooling Layer* 26](#_Toc56525528)

[*2.4.5 Fully Connected Layer* 27](#_Toc56525529)

[2.5 Arsitektur CNN 28](#_Toc56525530)

[2.5.1 Arsitektur *MobileNet* 28](#_Toc56525531)

[2.5.2 Arsitektur *InceptionV3* 31](#_Toc56525532)

[2.5.3 Arsitektur *VGG16* 32](#_Toc56525533)

[*2.6 Google Colaboratory* 34](#_Toc56525534)

[*2.7 Framework Keras dan Tensorflow* 35](#_Toc56525535)

[*2.8 CT Scan* 37](#_Toc56525536)

[*2.9 Coronavirus Disease 2019 (Covid-19)* 38](#_Toc56525537)

[BAB 3 METODE PENELITIAN 40](#_Toc56525538)

[3.1 Waktu dan Tempat 40](#_Toc56525539)

[3.2 Alat dan Bahan 40](#_Toc56525540)

[3.2.1 Perangkat Keras 40](#_Toc56525541)

[3.2.2 Perangkat Lunak 40](#_Toc56525542)

[3.2.3 Data Set 40](#_Toc56525543)

[3.3 Metode Penelitian 41](#_Toc56525544)

[3.3.1 Tahap Persiapan dan Pre Processing Dataset 43](#_Toc56525545)

[3.3.2 Tahap Desain Arsitektur 43](#_Toc56525546)

[3.3.3 Tahap Pengujian 44](#_Toc56525547)

[3.4 Waktu dan Jadwal Penelitian 44](#_Toc56525548)

[DAFTAR PUSTAKA 47](#_Toc56525549)

[LAMPIRAN 50](#_Toc56525550)

[Lampiran 1. Contoh Lampiran 1 50](#_Toc56525551)

[Lampiran 2. Contoh Lampiran 2 51](#_Toc56525552)

[BIODATA PENULIS 52](#_Toc56525553)

# DAFTAR TABEL

Tabel 2‑1Arsitektur MobileNet 30

Tabel 3‑1Rincian Jadwal Penelitian 44

# DAFTAR GAMBAR

Gambar 2‑1Perbedaan antara lapisan layer pada Jaringan Saraf Tiruan dengan lapisan layer deep learning 22

Gambar 2‑2Alur Proses CNN 23

Gambar 2‑3Proses Input Layer 24

Gambar 2‑4 Proses Convolutional Layer 25

Gambar 2‑5Fungsi Activation Layer 26

Gambar 2‑6Feature Map 27

Gambar 2‑7Proses Pooling 27

Gambar 2‑8 Proses Fully Conected Layer 28

Gambar 2‑9Arsitektur Depthwise Separable Convolutional 30

Gambar 2‑10Standard convolution (kiri), Depthwise separable convolution (kanan) 31

Gambar 2‑11Arsitektur VGG16 33

Gambar 2‑12Tampilan Antarmuka Google Colaboratory 35

Gambar 2‑13 Susunan Perangkat Keras dan Perangkat Lunak Deep Learning 36

Gambar 2‑14Citra CT Scan 37

# DAFTAR LAMPIRAN

Lampiran 1. Contoh Lampiran 1 37

Lampiran 2. Contoh Lampiran 2 38

# DAFTAR ISTILAH DAN SINGKATAN

# DAFTAR SIMBOL

# PENDAHULUAN

## Latar Belakang

Pada bulan Desember 2019 ditemukan wabah pneumonia yang terkait dengan virus korona baru dan disebut dengan sindrom pernafasan akut parah atau *Corona virus 2* (*SARS-Cov-2*) di Wuhan, China. Virus ini dapat menular begitu cepat, ketika seseorang terkena infeksi virus tersbeut maka akan mengalami beberapa gejala umum antara lain gejala gangguan pernapasan akut seperti demam, batuk, dan sesak napas. Kemudian WHO menamai penyakit yang disebabkan oleh *Novel Coronavirus* sebagai *Coronavirus Disease 2019* (*COVID-19*).[2]

Hingga saat ini proses untuk mendiagnosa dan konfirmasi *COVID-19* bergantung pada *real-time reverse-transcription–polymerase-chain-reaction (RT-PCR)* yang mendeteksi keberadaan *SARS-CoV-2*.[1] Selain konfirmasi *hasil RT-PCR*, elemen diagnostik utama lainnya yang dapat memfasilitasi identifikasi *COVID-19* adalah citra tomografi komputasi dada *(CT-Scan).* Karakteristik pencitraan dada *(CT-Scan)* dari paru-paru yang terinfeksi dilaporkan termasuk ke dalam *ground glass opacity (GGO)* dan konsolidasi berkorelasi berat yang dapat mengidentifikasi COVID-19.[3]

*Deep learning* adalah sub bidang pembelajaran khusus *dari machine learning* dan merupakan pandangan baru tentang pembelajaran dari data yang menekankan pada lapisan *(layers)* pembelajaran berturut-turut dari representasi yang semakin bermakna.[4] Model *deep learning* dapat mempelajari komputasinya sendiri atau dapat bekerja dengan otak sendiri. Sebuah model *deep learning* dirancang untuk terus menganalisa data dengan struktur logika yang menyerupai cara bagaimana manusia mengambil sebuah keputusan. *Deep learning* banyak diimplementasikan untuk klasifikasi maupun identifikasi suatu objek melalui citra atau gambar dengan cara mempelajari fitur dari data tersebut secara otomatis.

*Convolutional Neural Network* (CNN) adalah pengembangan dari *Multilayer Perceptron (MLP)* yang didesain untuk mengolah data dua dimensi, misalnya pada citra. CNNtermasuk dalam jenis *Deep Neural Network* karena kedalaman jaringan yang tinggi dan banyak diaplikasikan pada data citra.[5] Prinsip dari CNN adalah dapat belajar secara langsung dari data atau citra, sehingga dapat mengurangi beban pada pemrograman.

Oleh karena itu, untuk memberikan dasar bagi pemodelan sistemik yang dapat memfasilitasi diagnosis dini *COVID-19* dibutuhkan perancangan identifikasi *COVID-19* berdasarkan citra *CT-Scan* dengan menggunakan *deep learning*. Sehingga pada tugas akhir ini, penulis memilih bahan kajian tentang identifikasi *CT-Scan COVID-19* dengan judul **“KLASIFIKASI *CT SCAN COVID-19* BERDASARKAN CITRA DENGAN MENGGUNAKAN METODE *CONVOLUTIONAL NEURAL NETWORK (CNN)*”**

## Rumusan Masalah

1. Bagaimana perancangan arsitektur CNN untuk klasifikasi *lung parenchyma, risk, mortality dan Covid positivity berdassarkan CT Scan COVID-19*?
2. Bagaimana pelatihan untuk klasifikasi *lung parenchyma, risk, mortality dan Covid positivity berdassarkan CT Scan COVID-19?*
3. Bagaimana pengujian klasifikasi *lung parenchyma, risk, mortality dan Covid positivity berdassarkan CT Scan COVID-19* dengan metode CNN?

## Batasan Masalah

Batasan masalah yang dibuat membatasi fokus penelitian pada tugas akhir ini adalah sebagai berikut.

1. Metode deep learning yang digunakan untuk klasifikasi *CT-Scan COVID-19* dan *CT Scan* tingkat resiko *COVID-19* menggunakan *Convolutional Neural Network* (CNN)*.*
2. Program klasifikasi *CT-Scan COVID-19* berdasarkan citra dibuat dengan Bahasa Pemrograman *Python* dengan antarmuka dan infrastruktur *Google Colaboratory*
3. Pembuatan model *deep learning* pada klasifikasi *CT-Scan COVID-19* dengan citra menggunakan *Framework Keras* dan *TensorFlow*
4. Penentuan jumlah dataset pelatihan, validasi dan pengujian tidak dibahas dalam penelitian ini.

## Tujuan Penelitian

* 1. Merancang arsitektur CNNuntuk klasifikasi *lung parenchyma, risk, mortality dan Covid positivity berdassarkan CT Scan COVID-19.*
  2. Melatih arsitektur CNN untuk klasifikasi *lung parenchyma, risk, mortality dan Covid positivity berdassarkan CT Scan COVID-19.*
  3. Menguji arsitektur CNN klasifikasi *lung parenchyma, risk, mortality dan Covid positivity berdassarkan CT Scan COVID-19*
  4. Menggabungkan dua metode klasifikasi terbaik dan melakukan pengujian terhadap metode tersebut.
  5. Menggunakan model klasifikasi lung parenchyma dan risk classifier sebagai browser based classifier.

## Manfaat Penelitian

Manfaat yang diharapkan setelah penelitian ini selesai dilaksanakan adalah sebagai berikut.

1. Mampu menerapakan ilmu yang telah didapatkan pada mata kuliah yang bersangkutan untuk menyelesaikan tugas akhir.
2. Menawarkan solusi yang lebih murah dan efisien untuk klasifikasi *CT Scan COVID-19.*

## Sistematika Penulisan

Dalam penulisan laporan tugas akhir ini, dibuat sistematika penulisan sebagai berikut :

**BAB I Pendahuluan**

Bab Pendahuluan berisi tentang judul penelitian, latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, dan manfaat penelitian.

**BAB II Tinjauan Pustaka**

Bab Tinjauan Pustaka berisi tentang dasar kajian pustaka yang mendasari berbagai gagasan tentang penelitian terdahulu serta teori-teori pendukung untuk penelitian seperti penjelasan tentang pengolahan citra, *deep learning, convolutional neural network* (CNN), arsitektur CNN, *google colaboratory, framework keras* dan *tensorflow*, *CT Scan* dan *COVID-19*.

**BAB III Metode Penelitian**

Bab Metode Penelitian berisi tentang urutan langkah penelitian yang dilakukan, meliputi waktu dan tempat penelitian, alat dan bahan yang digunakan, metode penelitian, sumber data, alur penelitian dan jadwal penelitian.

**BAB IV Pembahasan**

Bab Pembahasan berisi tentang hasil dan analisa dari penelitian klasifikasi *CT Scan COVID-19* berdasarkan citra dengan menggunakan metode *convolutional neural network*. Pada bab ini juga akan membahas hasil kajian dan analisa yang disesuaikan dengan rumusan masalah yang dibuat.

**BAB V Penutup**

Bab Penutup berisi tentang kesimpulan dan saran atas hasil penelitian yang telah dilaksanakan.

# TINJAUAN PUSTAKA

## Penelitian Terdahulu

Dalam penyusunan penelitian tugas akhir “Klasifikasi CT Scan COVID-19 Berdasarkan Citra dengan Menggunakan Metode *Convolutional Neural Network*” terdapat beberapa penelitian terdahulu yang berhubungan yaitu membahas tantang klasifikasi objek berdasarkan citra dengan *machine learning*. Berikut ini adalah beberapa contoh dari penelitian tersebut :

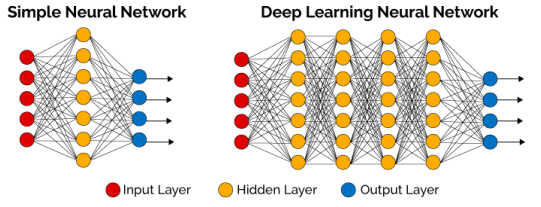
* 1. Penelitian Jepri dengan judul “Identifikasi Penyakit pada Daun Tomat dan Daun Singkong Berdasarkan Citra Daun Menggunakan Metode *Convolutional Neural Network* (CNN) Berbasis Android”[6]. Penelitian ini membahas tentang rancang bangun sistem berbasis *android* yang dapat mengklasifikasi penyakit pada daun tomat dan daun singkong berdasarkan citra dengan menggunakan arsitektur *VGG16*, *InceptionV3* dan *MobileNet*.
  2. Penelitian Andreas Galang Anugerah yang berjudul “Klasifikasi Tingkat Keganasan Kanker Paru-Paru pada Citra *Computed Tomography* (CT) *Scan* Menggunakan Metode *Convolutional Neural Network* (CNN)”[7]. Penelitian ini membahas tentang system klasifikasi tingkat keganasan kanker paru-paru berdasarkan citra *CT Scan* dengan menggunakan 4 arsitektur CNN yaitu *CanNet, LeNet, VGG16* dan *VGG19*.
  3. Penelitian Wanshan Ning, Shijun Lei, Jingjing Yang, Yukun Cao, Peiran Jiang, Qianqian ynag, Jiao Zhang, Xiaobei Wang, Fenghua Chen, Zhi Geng, Liang Xiong, Hongmei Zhou, Yaping Guo, Yulan Zeng, Heshui Shi, Lin Wang, Yu Xue dan Zheng Wang yang berjudul “*iCTCF: an Integrative Resource of Chest Computed Tomography Images and Clinical Features with COVID-19 Pneumonia”*[3]. Penelitian ini membahass tentang sistem yang dapat memprediksi dan mendiagnosa *COVID-19* dengan menggabungkan metode CNN untuk citra *CT Scan* dan metode DNN untuk *Clinical Features* (CF).

## Pengolahan Citra

Pengolahan citra adalah pemrosesan citra, khususnya dengan  
menggunakan komputer, menjadi citra yang kualitasnya lebih baik. Pengolahan  
citra bertujuan memperbaiki kualitas citra agar mudah diinterpretasi oleh manusia  
atau mesin (dalam hal ini komputer). Teknik-teknik pengolahan citra  
mentransformasikan citra menjadi citra lain. Jadi, masukannya adalah citra dan  
keluarannya juga citra yang berkualitas lebih baik daripada citra masukan.[8]

## *Deep Learning*

*Deep Learning* (Pembelajaran Dalam) atau sering dikenal dengan istilah  
Pembelajaran Struktural Mendalam (*Deep Structured Learning*) atau Pembelajaran Hierarki (*Hierarchical learning*) adalah salah satu cabang dari ilmu pembelajaran mesin (*Machine Learning*) yang terdiri algoritma pemodelan abstraksi tingkat tinggi pada data menggunakan sekumpulan fungsi transformasi non-linear yang ditata berlapis-lapis dan mendalam. Teknik dan algoritma dalam pembelajaran dalam dapat digunakan baik untuk kebutuhan pembelajaran terarah (*supervised learning*), pembelajaran tak terarah (*unsupervised learning*) dan semi-terarah (*semi-supervised learning*) dalam berbagai aplikasi seperti pengenalan citra, pengenalan suara, klasifikasi teks, dan sebagainya.[9]



Gambar 2‑1Perbedaan antara lapisan layer pada Jaringan Saraf Tiruan dengan lapisan layer deep learning

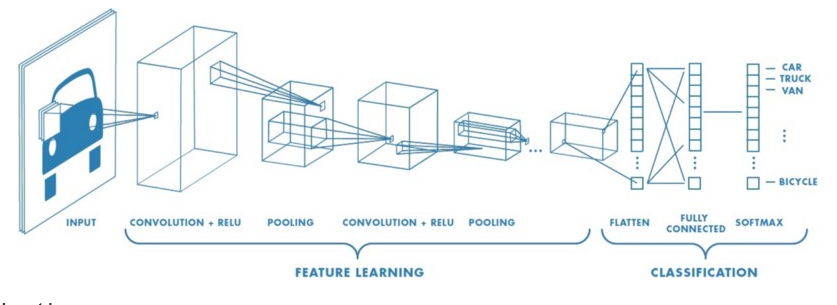
*Deep Learning* adalah salah satu jenis algoritma jaringan saraf tiruan yang menggunakan meta data sebagai *input* dan mengolahnya menggunakan sejumlah lapisan tersembunyi (*hidden layer*) transformasi non-linier dari data masukan untuk menghitung nilai *output*. Algoritma pada *Deep Learning* memiliki fitur yang unik yaitu sebuah fitur yang mampu mengekstraksi secara otomatis. Hal ini berarti algoritma yang dimilikinya secara otomatis dapat menangkap fitur yang relevan sebagai keperluan dalam pemecahan suatu masalah. Algoritma semacam ini sangat penting dalam sebuah kecerdasan buatan karena mampu mengurangi beban pemrograman dalam memilih fitur yang eksplisit.

Dalam jaringan saraf tiruan tipe *Deep Learning* setiap lapisan tersembunyi bertanggung jawab untuk melatih serangkaian fitur unik berdasarkan *output* dari jaringan sebelumnya. Algoritma ini akan menjadi semakin kompleks dan bersifat abstrak ketika jumlah lapisan tersembunyi (*hidden layer*) semakin bertambah banyak. Jaringan saraf yang dimiliki oleh *Deep Learning* terbentuk dari hierarki sederhana dengan beberapa lapisan hingga tingkat tinggi atau banyak lapisan(*multilayer*). Berdasarkan hal itulah *Deep Learning* dapat digunakan untuk memecahkan masalah kompleks yang lebih rumit dan terdiri dari sejumlah besar lapisan transformasi non linier.

## *Convolutional Neural Network*

*Convolutional Neural Network* (CNN) adalah pengembangan dari *Multilayer Perceptron* (MLP) yang didesain untuk mengolah data dua dimensi, misalnya pada citra. CNNtermasuk dalam jenis *Deep Neural Network* karena kedalaman jaringan yang tinggi dan banyak diaplikasikan pada data citra.[5]

Alur dari proses CNN dalam mengolah citra masukan hingga mengklasifikasikan citra tersbut ke dalam kelas tertentu berdasarkan nilai keluarannya dapat dilihat pada Gambar 2‑2



Gambar 2‑2Alur Proses CNN

### *Input Layer*

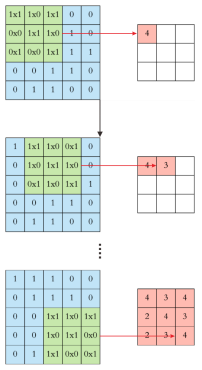
*Input Layer* padaCNN menampung nilai piksel dari citra masukan. Untuk citra dengan ukuran 64x64 dengan 3 channel warna, yaitu channel warna *RGB (Red, Green, Blue)*, maka yang menjadi masukan adalah piksel array yang memiliki ukuran 64x64x3.[10]

**

Gambar 2‑3Proses Input Layer

### *Convolutional Layer*

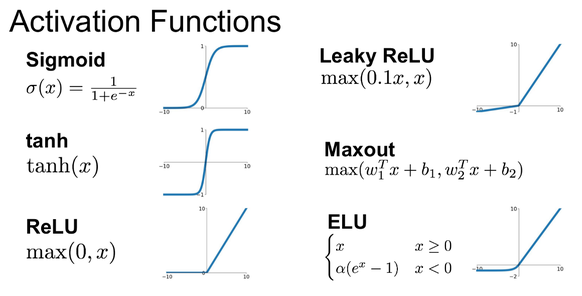
*Convolutional Layer* adalah inti dari CNN. *Convolutional Layer* menghasilkan citra baru yang menunjukkan fitur dari citra input. Dalam proses tersebut, *Convolutional Layer* menggunakan filter pada setiap citra yang menjadi masukan. Filter pada layer ini berupa array 2 dimensi yang memiliki ukuran 5x5, 3x3 atau 1x1. Proses *Convolutional Layer* dengan menggunakan filter pada layer ini akan menghasilkan *feature map* yang akan digunakan pada *Action Layer*. Gambar 2‑4 menunjukkan alur proses dari *Convolutional Layer*



Gambar 2‑4 Proses Convolutional Layer

### *Activation Layer*

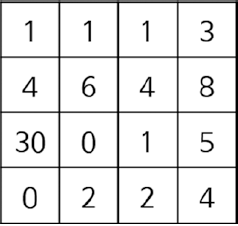
*Activation Layer* adalah layer dimana *feature map* dimasukkan ke dalam fungsi aktivasi. Fungsi aktivasi digunakan untuk mengubah nilai-nilai pada *feature map* pada *range* tertentu sesuai dengan fungsi aktivasi yang digunakan. Ini bertujuan untuk meneruskan nilai yang menampilkan fitur dominan dari citra yang masuk pada layer berikutnya.[10] Fungsi aktivasi yang umum digunakan dapat dilihat pada Gambar 2‑5



Gambar 2‑5Fungsi Activation Layer

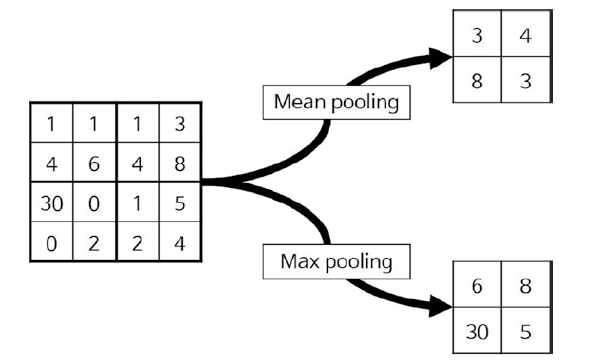
### *Pooling Layer*

*Pooling Layer* menerima masukan dari *activation layer* kemudian mengurangi jumlah parameternya. *Pooling* juga biasa disebut dengan *subsampling* atau *downsampling* yang mengurangi dimensi dari *feature map* tanpa menghilangkan informasi penting di dalamnya. Proses dalam *Pooling Layer* cukup sederhana, pertama proses akan menentukan *downsampling* yang akan digunakan pada *feature map*, misalnya 2x2. Setelah itu proses akan melakukan *pooling*  pada *feature map*, sebagai contoh akan digunakan *feature map* berukuran 4x4 berikut.



Gambar 2‑6Feature Map

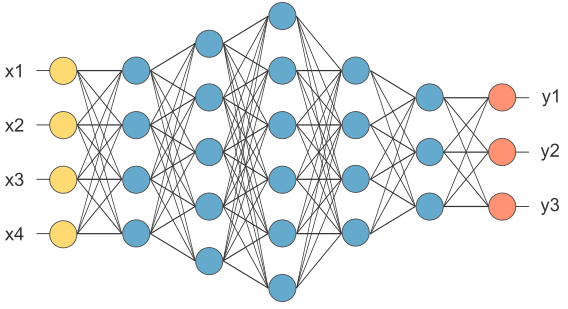
Kemudian selanjutnya proses akan menggunakan matriks 2x2 untuk melakukan *pooling*. Proses *pooling* sendiri terdiri dari beberapa macam seperti *Max Pooling*, *Mean Pooling* dan *Sum Pooling*.



Gambar 2‑7Proses Pooling

### *Fully Connected Layer*

Setelah melalui beberapa proses di atas, kemudian hasil dari *pooling layer* akan digunakan sebagai masukan untuk *Fully Connected Layer*. Layer ini memiliki kesamaan struktur dengan *Artificial Neural Network* pada umumnya yaitu memiliki *input layer*, *hidden layer* dan *output layer* dimana masing-masing memiliki neuron-neuron yang saling terhubung dengan neuron-neuron pada layer tetangganya. Gambar 2‑8 menunjukkan contoh dari proses *Fully Connected Layer*



Gambar 2‑8 Proses Fully Conected Layer

Pada Gambar 2‑8 dapat diketahui bahwa sebelum hasil *pooling* digunakan sebagai masukan, terlebih dahulu diubah menjadi vector (x1, x2, x3, dst) kemudian selanjutnya akan diproses ke dalam *Fully Connected Layer*. Pada layer terakhir di dalam *Fully Connected Layer* akan digunakan fungsi aktivasi *sigmoid* atau *softmax* untuk menentukan klasifikasi dari citra masukan dari *input layer CNN*.

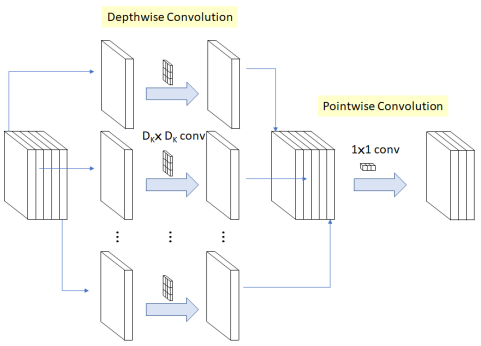
## Arsitektur CNN

### Arsitektur *MobileNet*

*MobileNet* merupakan salah satu arsitektur *Convolutional Neural Network* (CNN) yang dapat digunakan untuk mengatasi kebutuhan akan *computing resource* berlebih. Sama seperti namanya, *MobileNet* dibuat agar dapat digunakan untuk ponsel[11]. Perbedaan mendasar antara arsitektur *MobileNet* dan arsitektur CNN pada umumnya adalah penggunaan lapisan atau layer konvolusi dengan ketebalan filter yang sesuai dengan ketebalan citra masukan. *MobileNet* membagi konvolusi menjadi *depthwise convolution* dan *pointwise convolution*.

* + 1. *Depthwise Separable Convolution*

Model *MobileNet* didasarkan pada konvolusi mendalam yang dapat  
dipisahkan (*depthwise separable convolution*), yaitu bentuk konvolusi yang dibentuk dengan menguraikan konvolusi standar (*standard convolution*) menjadi konvolusi mendalam (*depthwise convolution*) dan konvolusi 1×1 yang disebut konvolusi searah (*pointwise convolution*) [12]. Untuk *MobileNets*, *depthwise convolution* menerapkan satu *filter* ke setiap saluran masukkannya. *Pointwise convolution* kemudian menerapkan konvolusi 1×1 untuk menggabungkan keluaran dari *depthwise convolution*. Konvolusi standar melakukan *filter* dan menggabungkan masukan ke dalam *set* keluaran baru dalam satu langkah. *Depthwise separable convolution* membaginya menjadi dua lapisan, lapisan terpisah untuk *filter* dan lapisan terpisah untuk menggabungkannya. Penguraian ini memiliki efek mengurangi komputasi dan ukuran model secara drastis.[12] Arsitektur dari *Depthwise separable convolution* dapat dilihat pada Gambar 2‑9



Gambar 2‑9Arsitektur Depthwise Separable Convolutional

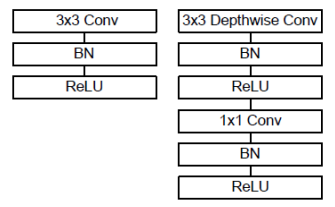
* + 1. Struktur Jaringan

Arsitektur *MobileNet* didefinisikan seperti pada Tabel 2‑1 . Semua lapisan diikuti oleh *batchnorm* (*backnormalization*) dan *ReLU non-linier* dengan pengecualian lapisan akhir yang terhubung penuh yang tidak memiliki nonlinier dan dimasukkan ke dalam lapisan *softmax* untuk klasifikasi.

Tabel 2‑1Arsitektur MobileNet

|  |  |  |
| --- | --- | --- |
| *Type/Stride* | *Filter Shape* | *Input Size* |
| Conv / s2 | 3 × 3 × 3 × 32 | 224 × 224 × 3 |
| Conv dw / s1 | 3 × 3 × 32 dw | 112 × 112 × 32 |
| Conv / s1 | 1 × 1 × 32 × 64 | 112 × 112 × 32 |
| Conv dw / s2 | 3 × 3 × 64 dw | 112 × 112 × 64 |
| Conv / s1 | 1 × 1 × 64 × 128 | 56 × 56 × 64 |
| Conv dw / s1 | 3 × 3 × 128 dw | 56 × 56 × 128 |
| Conv / s1 | 1 × 1 × 128 × 128 | 56 × 56 × 128 |
| Conv dw / s2 | 3 × 3 × 128 dw | 56 × 56 × 128 |
| Conv / s1 | 1 × 1 × 128 × 256 | 28 × 28 × 128 |
| Conv dw / s1 | 3 x 3 x 256 dw | 28 x 28 x 256 |
| Conv / s1 | 1 x 1 x 256 x 256 | 28 x 28 x 256 |
| Conv dw / s2 | 3 x 3 x 256 dw | 28 x 28 x 256 |
| Conv / s1 | 1 × 1 × 256 × 512 | 14 × 14 × 256 |
| 5x Conv dw / s1 | 3 × 3 × 512 dw | 14 × 14 × 512 |
| 5x Conv / s1 | 1 × 1 × 512 × 512 | 14 × 14 × 512 |
| Conv dw / s2 | 3 × 3 × 512 dw | 14 × 14 × 512 |
| Conv / s1 | 1 × 1 × 512 × 1024 | 7 × 7 × 512 |
| Conv dw / s2 | 3 × 3 × 1024 dw | 7 × 7 × 1024 |
| Conv / s1 | 1 × 1 × 1024 × 1024 | 7 × 7 × 1024 |
| Avg Pool / s1 | Pool 7 × 7 | 7 × 7 × 1024 |
| FC / s1 | 1024 × 1000 | 1 × 1 × 1024 |
| Softmax / s1 | Classifier | 1 × 1 × 1000 |

Pada sebuah lapisan konvolusi standar hanya menggunakan lapisan  
konvolusi biasa sebesar 3x3. Sedangkan pada sebuah lapisan *depthwise separable  
convolution* dipisahkan menjadi dua konvolusi yaitu 3x3 *depthwise convolution* dan 1x1 *pointwise convolution* serta *batchnorm* dan *ReLU non-linier* di setiap lapisan konvolusinya seperti pada Gambar 2‑10 berikut.



Gambar 2‑10Standard convolution (kiri), Depthwise separable convolution (kanan)

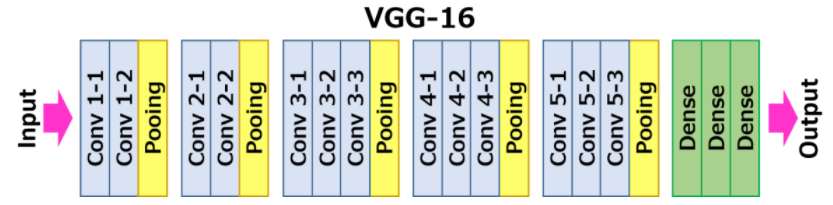
### Arsitektur *InceptionV3*

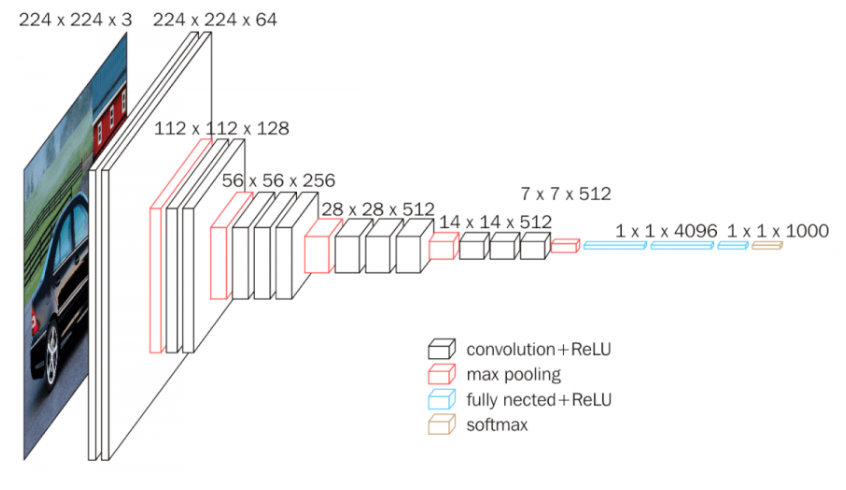
*InceptionV3* adalah sebuah model *deep learning convolutional neural  
network* yangdikembangkan oleh *Google* untukmemenuhi *ImageNet Large Visual* *recognition challenge* pada tahun 2012. Model *Inception* menggunakan filter pada layer *convolutional,* tidak seperti *convolutional layer* biasa. Hasil dari beberapa filter tersebut dijadikan satu lagi menggunakan *channel concat* sebelum masuk ke dalam iterasi berikutnya[13].

Tujuan dari modul *inception* adalah untuk bertindak sebagai *multilevel  
feature extractor* dengan menghitung filter-filter *convolutional* dalam modul yang  
sama. Hasil dari filter-filter tersebut kemudian ditumpukan ke dalam dimensi  
*channel* sebelum dimasukan ke dalam lapisan selanjutnya.

### Arsitektur *VGG16*

*VGG16* adalah model jaringan saraf konvolusional yang diusulkan oleh K. Simonyan dan A. Zisserman dari *Universitas Oxford* dalam makalahnya yang berjudul "*Very Deep Convolutional Networks for Large-Scale Image Recognition "*.[14] Model ini mencapai top-5 dengan akurasi pengujian 92,7% di *ImageNet*, yang memiliki kumpulan data lebih dari 14 juta gambar dan 1000 kelas. *VGG16* adalah salah satu model terkenal yang dikirim ke *ILSVRC-2014*. Gambar 2‑9 berikut menunjukkan arsitektur *VGG16*.





Gambar 2‑11Arsitektur VGG16

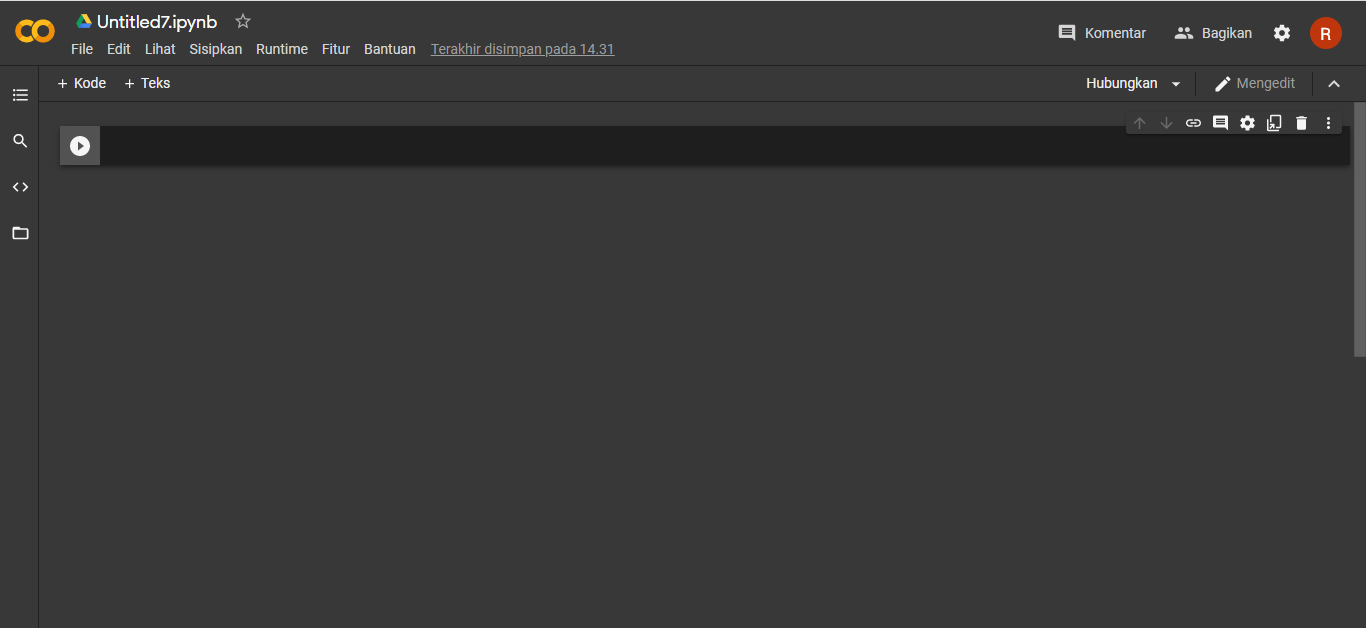
Input ke layer cov1 berupa citra berukuran tetap 224 x 224 RGB. Citra melewati tumpukan lapisan convolusional (conv.), Di mana filter digunakan dengan bidang penerima yang sangat kecil sebesar 3×3. Dalam salah satu konfigurasinya juga menggunakan filter konvolusi 1 × 1, yang dapat dilihat sebagai transformasi linier dari saluran input (diikuti oleh non-linieritas). Langkah (Stride) konvolusi ditetapkan ke 1 piksel; spasial padding konv. input lapisan sedemikian rupa sehingga resolusi spasial dipertahankan setelah konvolusi, yaitu padding adalah 1-piksel untuk konv. 3 × 3. lapisan. Penyatuan spasial dilakukan oleh lima lapisan max-pooling, yang mengikuti beberapa konv. lapisan (tidak semua lapisan konv. diikuti oleh max pooling). Max-pooling dilakukan melalui jendela 2x2 piksel, dan dengan 2 stride.

Tiga lapisan Fully-Connected (FC) mengikuti tumpukan lapisan convolusional (yang memiliki kedalaman berbeda dalam arsitektur berbeda) dua yang pertama memiliki masing-masing 4096 kanal, yang ketiga melakukan klasifikasi ILSVRC 1000 arah dan karenanya berisi 1000 kanal (satu untuk masing-masing kelas). Lapisan terakhir adalah lapisan soft-max. Konfigurasi dari lapisan Fully-Connected sama di semua jaringan.

Semua hidden layers dilengkapi dengan fungsi aktivasi ReLU. Juga dicatat bahwa tidak ada jaringan (kecuali satu) yang berisi Local Response Normalization (LRN), normalisasi tersebut tidak meningkatkan kinerja pada dataset ILSVRC, namun mengarah pada peningkatan konsumsi memori dan waktu komputasi.

## *Google Colaboratory*

*Google Colaboratory* adalah layanan berbasis *cloud Google* yang mereplikasi *Jupyter Notebook* pada *cloud*. Dengan *Google Colaboratory* tidak perlu menginstal apapun untuk menggunakannya. Pada sebagian besar hal, penggunaan *Google Colaboratory* sama seperti instalasi *pada Desktop Jupyter Notebook*.[15] *Google Colaboratory* ditujukan khusus untuk para pembaca yang menggunakan sesuatu selain pengaturan desktop standar untuk mengerjakan contoh - contoh. Gambar 2‑9 menunjukkan tampilan antarmuka untuk Google Colaboratory.



Gambar 2‑12Tampilan Antarmuka Google Colaboratory

Untuk menggunakan Google Colaboratory, diwajibkan harus memiliki akun Google untuk dapat mengakses agar sebagian besar fitur pada Google Colaboratory dapat berfungsi dengan baik.

Seperti halnya Jupyter Notebook, penggunaan Google Colaboratory untuk melakukan tugas tertentu dalam paradigma berorientasi sel. Terdapat kemiripan pada antarmuka Jupyter Notebook dan Google Colaboratory. Google Colaboratory dapat membuat berbagai tipe sel dan menggunakannya untuk membuat buku catatan.

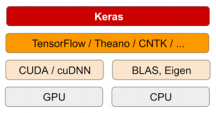
## *Framework Keras dan Tensorflow*

*Keras* adalah salah satu *framework deep learning* untuk *Python* yang mampu melatih dan mendefinisikan hampir semua jenis model *deep learning*.[4] *Keras* menawarkan *Application Programming Interface* (API) yang konsisten dan mampu berjalan di atas *Tensorflow, Theano* atau *Microsoft Cognitive Toolkit* (CNTK).

*Keras* memiliki beberapa fitur utama sebagai berikut :

* *Keras* memungkinkan kode yang sama untuk berjalan mulus di CPU atau GPU
* Memiliki API yang ramah pengguna dan memudahkan pembuatan prototipe model pembelajaran dalam dengan cepat
* Memiliki dukungan built-in untuk jaringan konvolusional (untuk *computer vision*), jaringan berulang (untuk pemrosesan urutan), dan kombinasi antara keduanya
* Mendukung arsitektur jaringan arbitrer: model multi-input atau multi-output, berbagi lapisan, berbagi model, dan sebagainya. *Keras* pada dasarnya cocok untuk membuat model pembelajaran dalam apa pun.

*Keras* didistribusikan di bawah lisensi MIT, yang berarti dapat digunakan secara bebas dalam proyek komersial. *Keras* kompatibel dengan semua versi *Python* dari 2.7 hingga 3.6.



Gambar 2‑13 Susunan Perangkat Keras dan Perangkat Lunak Deep Learning

Gambar 2‑10 merupakan susunan perangkat keras dan perangkat lunak deep learning. Melalui TensorFlow (atau Theano, atau CNTK), Keras dapat berjalan mulus di CPU dan GPU. Saat berjalan di CPU, TensorFlow sendiri menggabungkan library level rendah untuk operasi tensor yang disebut Eigen. Pada GPU, TensorFlow menggabungkan pustaka operasi deep learning yang dioptimalkan dengan baik yang disebut NVIDIA CUDA Deep Neural Network library (cuDNN).

### *Tensorflow.js*

Tensorflow.js merupakan tool Javascript library yang dikembangkan oleh Google dan diperkenalkan sebagai open-source pada tahun 2015, tool ini digunakan untuk training dan penggunaan machine learning model di browser. Tensorflow.js merupakan library tambahan untuk Tensorflow yang merupakan library dari machine learning untuk phyton. [16]

*TensorFlow.js* menyediakan banyak fitur dan model yang akan  
menyederhanakan tugas yang sulit dan menghabiskan waktu untuk dilatih model  
*machine learning*. Di sisi model, *TensorFlow.js* hadir dengan beberapa model  
pelatihan untuk tujuan yang berbeda, seperti *PoseNet* untuk memperkirakan pose  
yang dilakukan seseorang secara *real time*. *Toxicity classifier*, untuk mendeteksi  
apakah suatu teks mengandung konten *toxic*, dan yang terakhir model *Coco SSD*,  
model yang digunakan untuk mendeteksi objek yang mengidentifikasi dan  
melokalkan beberapa objek dalam sebuah gambar.

## *Website*

*Website* adalah halaman informasi yang disediakan melalui jalur internet sehingga bissa diakses selama terkoneksi dengan jaringan internet. Sebuah situs website umumnya merupakan bagian dari suatu nama domain atau sub domain di *World Wide Web* (WWW) di internet. WWW terdiri dari seluruh situs web yang tersedia kepada publik. Halaman – halaman sebuah situs web diakses dari sebuah URL yang menjadi akar (*root*) yang disebut dengan *homepage* (beranda).[17]

Sebuah halaman web adalah dokumen yang ditulis dalam format HTML (*Hyper Text Markup Language*) yang hampir selalu bias diakses melalui HTTP. HTTP yaitu protokol yang menyampaikan informasi dari *server website* untuk ditampilkan kepada *users* melalui web browser.

Beberapa website membutuhkan subskripsi (data masukan) agar user dpaat mengakses sebagian atau seluruh isi dari website tersebut. Misalnya pada beberapa situs bisnis yang membutuhkan subskripsi agar user dapat mengaksesnya.

### HTML (*Hyper Text Markup Language*)

Hypertext Markup Language (HTML) adalah bahasa markup yang umum digunakan untuk membuat halaman web. Sebenarnya HTML bukanlah sebuah bahasa pemrograman. Jika ditinjau dari namanya, HTML merupakan bahasa markup atau penandaan terhadap sebuah dokumen teks. Tanda tersebut di gunakan untuk menentukan format atau style dari teks yang ditandai.[18]

HTML dibuat oleh Tim Berners-Lee ketika masih bekerja untuk CERN dan dipopulerkan pertama kali oleh browser Mosaic. Selama awal tahun 1990 HTML mengalami perkembangan yang sangat pesat. Setiap pengembangan HTML pasti akan menambahkan kemampuan dan fasilitas yang lebih baik dari versi sebelumnya. Sebelum suatu HTML disahkan sebagai suatu dokumen HTML standar, ia harus disetujui dulu oleh W3C untuk dievaluasi secara ketat. Setiap terjadi perkembangan suatu versi HTML, maka mau tak mau browser pun harus memperbaiki diri agar bisa mendukung kode-kode HTML yang baru tersebut. Sebab jika tidak, browser tak akan bisa menampilkan HTML tersebut.

### CSS (*Cascading Style Sheets*)

CSS adalah bahasa penulisan yang digunakan untuk mendeskripsikan penampilan dari sebuah dokumen *markup*. Dalam penggunaannya CSS selalu digunakan bersama HTML. CSS biasanya disimpan di dalam sebuah file berekstensi .css dan disematkan di dalam dokumen HTML untuk memberikan *style* pada halaman tersebut. Pada dasarnya CSS dibuat untuk memisahkan *style* dari halaman web, sehingga antara konten pada HTML dan desain tampilan pada dokumen CSS dapat dikerjakan di dua tempat berbeda.[19]

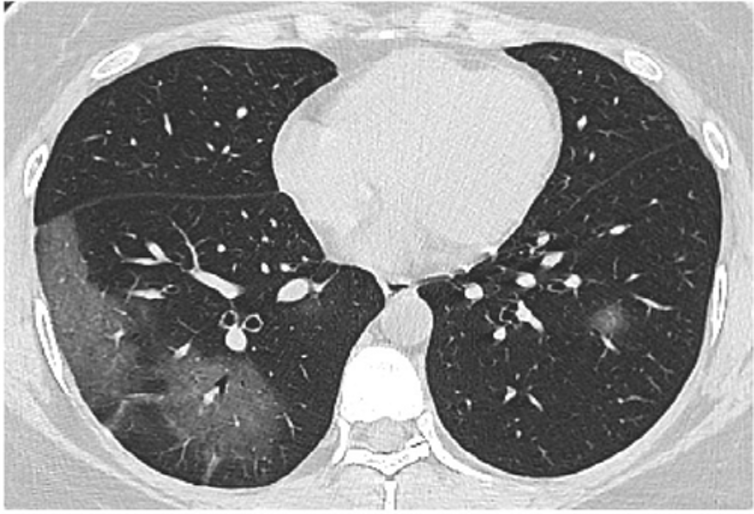
### *Javascript*

*JavaScript* adalah bahasa pemrograman web yang bersifat *Client Side Programming Language*. *Client Side Programming Language* adalah tipe bahasa pemrograman yang pemrosesannya dilakukan oleh *client*. Aplikasi *client* yang dimaksud merujuk kepada *web browser* seperti *Google Chrome, Mozilla Firefox, Opera Mini* dan sebagainya. [20]

*JavaScript* pertama kali dikembangkan pada pertengahan dekade 90-an. *JavaScript* berbeda dengan bahasa pemrograman *Java*. Untuk penulisannya, *JavaScript* dapat disisipkan di dalam dokumen HTML ataupun dijadikan dokumen tersendiri yang kemudian diasosiasikan dengan dokumen lain yang dituju. *JavaScript* mengimplementasikan fitur yang dirancang untuk mengendalikan bagaimana sebuah halaman web berinteraksi dengan *user*.

## CT Scan

*CT Scan* adalah Pemindaian tomografi yang menggabungkan serangkaian gambar sinar-X yang diambil dari berbagai sudut di sekitar tubuh dan menggunakan pemrosesan komputer untuk membuat gambar penampang (irisan) tulang, pembuluh darah, dan jaringan di dalam tubuh[21]. Gambar *CT scan* memberikan informasi yang lebih rinci daripada sinar-X biasa.



Gambar 2‑14Citra CT Scan

*CT scan* memiliki beberapa kegunaan. *CT scan* dapat digunakan untuk memvisualisasikan hampir semua bagian tubuh dan digunakan untuk mendiagnosis penyakit atau cedera serta untuk merencanakan perawatan medis, bedah atau radiasi.

## *Coronavirus Disease 2019 (Covid-19)*

*Coronavirus Disease 2019 (COVID-19)* adalah penyakit saluran pernafasan akut yang disebabkan oleh *Severe Acute Respiratory Syndrome Coronavirus 2 (SARS-COV2)* dan dapat menular[22]. *COVID-19* pertama ditemukan pada Desember 2019 di Wuhan, China. Selanjutnya tanggal 30 Januari 2020, *World Health Organization* (WHO) menetapkan kejadian tersebut sebagai Kedaruratan Kesehatan Masyarakat yang Meresahkan Dunia (KKMMD) dan pada tanggal 11 Maret 2020 *COVID-19* ditetapkan sebagai pandemi.

Bentuk gejala umum dari *COVID-19* berupa gejala pernafasan akut seperti demam , batuk kering dan sesak nafas. Dengan kasus yang berat akan menyebabkan pneumonia, sindrom pernafasan akut, gagal ginjal, hingga kematian.

Sampai sekarang untuk mendeteksi *COVID-19* menggunakan tes amplifikasi *asam nukleat* (NAAT) seperti *real time reversetranscription polymerase chain reaction (*rRT-PCR*)*[23]. Selain itu, ada beberapa elemen diagnostik utama lainnya yang dapat memfasilitasi identifikasi *COVID-19* termasuk fitur klinis (CF) dan pencitraan tomografi komputasi dada (*CT Scan* ) . Karakteristik pencitraan *CT* Scan dari paru-paru yang terinfeksi dilaporkan termasuk *ground glass opacity (*GGO*)* dan konsolidasi berkorelasi berat. Meskipun gambaran lengkap mengenai masing-masing elemen masih menunggu penggambaran penuh, penggabungan fitur elemen diagnostik tersebut di atas secara komprehensif dapat secara kolektif meningkatkan akurasi dan kemanjuran diagnosis.[3]

# METODE PENELITIAN

## Waktu dan Tempat

Penelitian ini dilakukan pada bulan Desember 2020 sampai bulan Maret 2021 di Kampus Fakultas Teknik Universitas Jenderal Soedirman yang terletak di Km5. Jl Mayor Jenderal Sungkono, Desa Blater, Kabupaten Purbalingga, Jawa Tengah.

## Alat dan Bahan

Alat dan bahan yang digunakan pada penelitian tugas akhir ini adalah sebagai berikut.

### Perangkat Keras

* 1. Laptop *ASUS M409DA* dengan spesifikasi *processor AMD Athlon Silver 3050U.*

### Perangkat Lunak

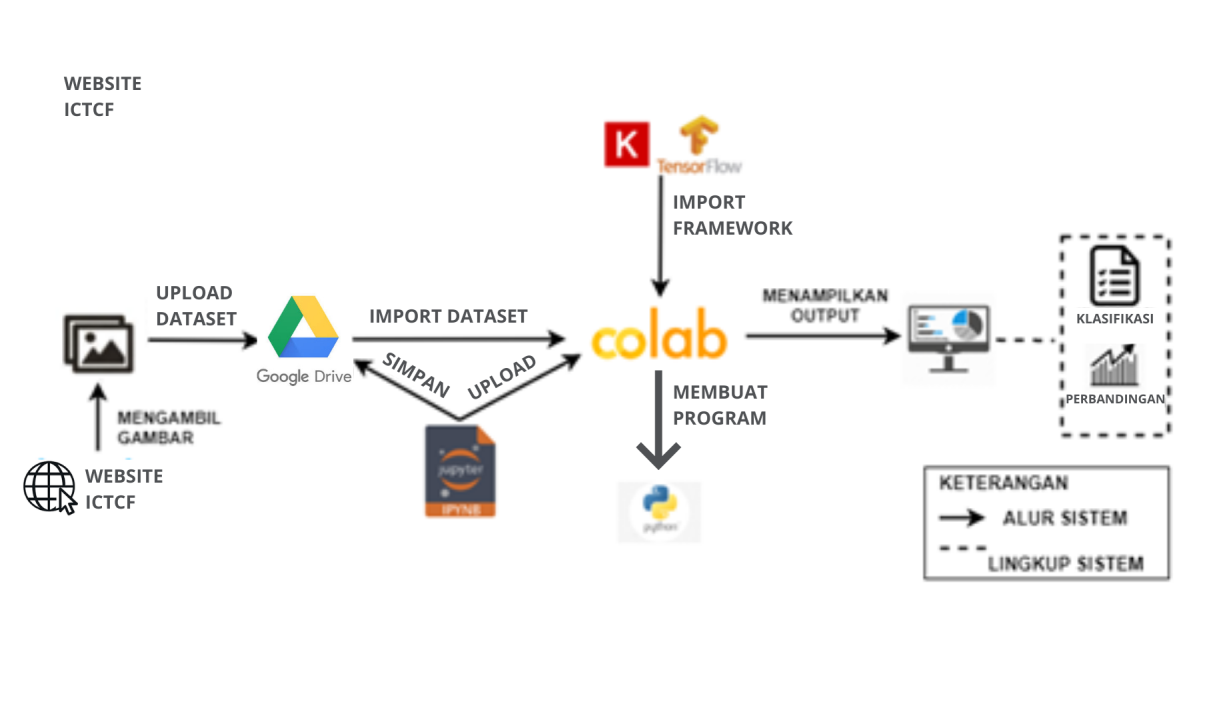
* + 1. Sistem Operasi *Windows 10 64 bit*
    2. Web browser *Google Chrome Version* 87.0.4280.88 (64-bit)
    3. *Google Colaboratory*
    4. *Google Drive*
    5. Layanan *Repository Web Development* pada platform *Github*

### Data Set

Pada penelitian ini digunakan data set untuk melakukan *training* pada proses *Deep Learning* yang diperoleh dari kumpulan data set *CT Scan Covid-19* di website iCTCFdalam bentuk gambar \*.jpg

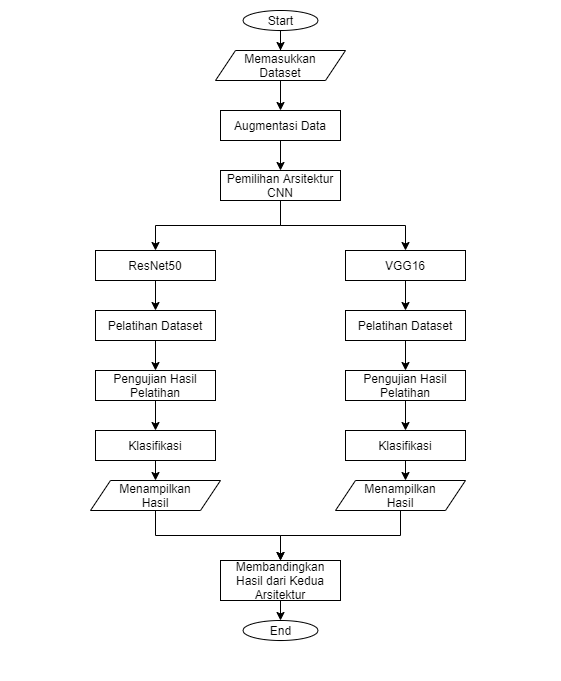
## Metode Penelitian

Penelitian ini dilaksanakan untuk melakukan empat klasifikasi, yaitu *lung parenchyma classifier, risk classifier, mortality classifier* dan *Covid positivity classifier* melalui CT Scan. Penelitian ini dibagi ke dalam beberapa tahapan yaitu tahap persiapan dan *pre-processing* data set, tahap desain arsitektur, dan tahap pengujian. Desain arsitektur dari sistem yang akan dibuat disajikan pada Gambar 3‑1 berikut :



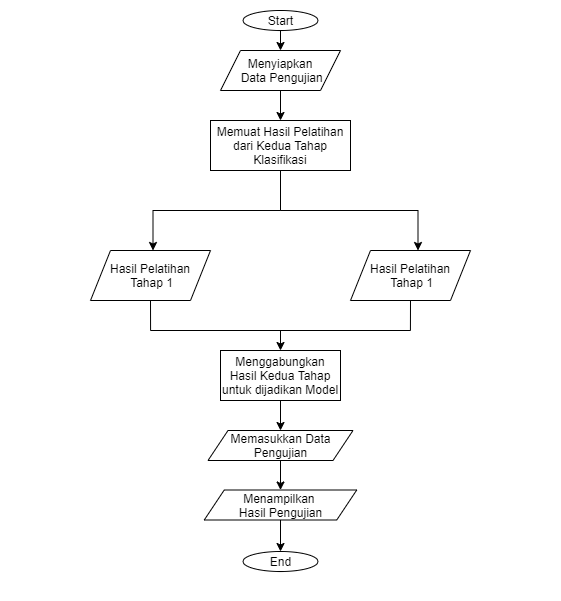
Gambar 3‑1Desain arsitektur sistem

Kemudian untuk diagram alir masing – masing sistem klasifikasi pada penelitian ini dapat dilihat pada Gambar 3‑2 berikut :



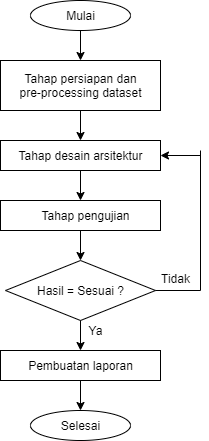
Gambar 3‑3Diagram alir sistem klasifikasi

Untuk diagram alir pengujian sistem yang merupakan gabungan dari dua tahap dapat dilihat seperti pada Gambar 3‑3 berikut :



Gambar 3‑4Diagram alir pengujian

Adapun diagram alir dari tahapan-tahapan penelitian yang dilakukan adalah seperti yang disajikan pada Gambar 3‑2berikut.



Gambar 3‑5Diagram alir penelitian

### Tahap Persiapan dan *Pre-Processing* Dataset

Pada tahap ini penyiapan dan *pre-processing* *dataset* dilakukan dengan mengambil gambar dataset  *CT Scan COVID-19* dengan format gambar “\*.jpg” dari Kaggle dan web iCTCFyang selanjutnya akan dimasukkan ke *Google Colaboratory*.

### Tahap Desain Arsitektur

Proses awal desain arsitektur dimulai dari membuat kode sumber untuk program CNN dan mengimpor *Framework Keras* dan *TensorFlow* yang dibuat menggunakan infrastruktur *Google Colaboratory* dengan bahasa pemrograman *Python* dan disimpan dalam bentuk *file Jupyter Notebooks* “.ipynb” dan kemudian disimpan ke *Github*. Selanjutnya *dataset* pelatihan tersebut diambil dari *Google Drive* dan disimpan ke dalam tempat penyimpanan sementara pada *Google Colaboratory*. Sistem yang dirancang ini menggunakan dua macam arsitektur jaringan CNN, yaitu *ResNet50* dan *VGG16* untuk masing-masing tahap klasifikasi berdasarkan *CT Scan* sedangkan untuk metode yang digunakan adalah metode *transfer learning* dengan mengambil *file Keras* “\*.h5” yang sudah dilatih sebelumnya pada kedua arsitektur tersebut dan melakukan *feature extraction* terhadap *file* tersebut. Selain itu juga ditambah satu *filter*/*feature map* tambahan sebagai keluaran untuk klasifikasi tersebut.

### Tahap Pengujian

Pada tahap pengujian seluruh datasetpelatihan dilatih terhadap masing- masing arsitektur jaringan CNN yang digunakan sampai dengan panjangnya *epochs* yang ditentukan. Sesudah itu datasetpengujian diekstrak dan dibandingkan hasilnya dengan data pengujian untuk melakukan prediksi pada setiap tahap klasifikasi serta membandingkan hasilnya dengan kedua arsitektur yang digunakan.

Selain itu tahap pengujian juga dilakukan dengan menggabungkan dua tahap klasifikasi dengan mengambil unjuk kerja terbaik dari masing-masing tahap klasifikasi. Kemudian dari kedua tahap tersebut akan digabungkan dan dilakukan pengujian.

## Waktu dan Jadwal Penelitian

Penelitian dilaksanakan dalam waktu 4 bulan dimulai dari bulan Desember 2020 sampai dengan bulan Maret 2021 dengan rincian jadwal kegiatan sebagai berikut.

Tabel 3‑1Rincian Jadwal Penelitian

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No | Kegiatan | Desember | | | | Januari | | | | Februari | | | | Maret | | | |
| 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 |
| 1. | Studi Pustaka |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2. | Persiapan dan *pre-processing* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3. | Desain arsitektur |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4. | Pengujian sistem |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5. | Evaluasi sistem |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6. | Pembuatan laporan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

# DAFTAR PUSTAKA

[1] Q. Li *et al.*, “Early Transmission Dynamics in Wuhan, China, of Novel Coronavirus–Infected Pneumonia,” *N. Engl. J. Med.*, vol. 382, no. 13, pp. 1199–1207, 2020, doi: 10.1056/nejmoa2001316.

[2] Z. Y. Zu *et al.*, “H13. Coronavirus Disease 2019 (COVID-19): A Perspective from ChinaZu, Z. Y., Jiang, M. D., Xu, P. P., Chen, W., Ni, Q. Q., Lu, G. M., & Zhang, L. J. (2020). H13. Coronavirus Disease 2019 (COVID-19): A Perspective from China. Radiology, 200490. https://doi,” *Radiology*, vol. 2019, p. 200490, 2020.

[3] W. Ning *et al.*, *iCTCF: an integrative resource of chest computed tomography images and clinical features of patients with COVID-19 pneumonia*. 2020.

[4] F. Chollet, *Deep Learning with Python*. Shelter Island: Manning Publications Co, 2017.

[5] W. S. Eka Putra, “Klasifikasi Citra Menggunakan Convolutional Neural Network (CNN) pada Caltech 101,” *J. Tek. ITS*, vol. 5, no. 1, 2016, doi: 10.12962/j23373539.v5i1.15696.

[6] Jepri, “IDENTIFIKASI PENYAKIT PADA DAUN TOMAT DAN DAUN SINGKONG BERDASARKAN CITRA DAUN MENGGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK (CNN) BERBASIS ANDROID,” 2019.

[7] A. G. Anugerah and D. Informatika, “KLASIFIKASI TINGKAT KEGANASAN KANKER PARU-PARU PADA COMPUTED TOMOGRAPHY ( CT ) SCAN MENGGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK,” 2018.

[8] R. Munir, *Pengolahan Citra Digital dengan Pendekatan Algoritmik*. Bandung: Informatika, 2004.

[9] Xenonstack, “Log Analytics With Deep Learning And Machine Learning,” 2017. https://medium.com/@xenonstack/loganalytics-with-deep-learning-and-machine-learning-20a1891ff70e (accessed Nov. 11, 2020).

[10] Sofyan, “Pengenalan Convolutional Neural Network – Part 1,” 2019. http://sofyantandungan.com/pengenalan-convolutional-neural-network-part-1/ (accessed Nov. 11, 2020).

[11] R. Okta, “MobileNet: Deteksi Objek pada Platform Mobile,” 2018. https://medium.com/nodeflux/mobilenet-deteksi-objek-pada-platform-mobile-bbbf3806e4b3 (accessed Nov. 15, 2020).

[12] A. G. Howard *et al.*, “MobileNets: Efficient convolutional neural networks for mobile vision applications,” *arXiv*, no. April 2017, 2017.

[13] S. Materials, K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition A. Object Detection Baselines,” pp. 9–12, 2015, [Online]. Available: http://host.robots.ox.ac.uk:8080/anonymous/3OJ4OJ.html,.

[14] Neurohive, “VGG16 – Convolutional Network for Classification and Detection,” 2018. https://neurohive.io/en/popular-networks/vgg16/ (accessed Nov. 15, 2020).

[15] J. Paul Mueller and L. Massaron, “What is Google Colaboratory?” https://www.dummies.com/programming/python/working-with-google-colaboratory-notebooks/ (accessed Nov. 15, 2020).

[16] A. Rahman, “Belajar Tensorflow.js,” 2019. https://alfian-pesan.medium.com/belajar-tensorflow-js-a9753f1237a1#:~:text=TensorFlowJs ini merupakan tool JavaScript,merupakan ML library untuk phyton. (accessed Feb. 21, 2021).

[17] R. Hidayatullah, “Pembuatan Desain Website Sebagai Penunjang Company Profile CV. Hensindo.,” pp. 11–25, 2016, [Online]. Available: http://sir.stikom.edu/id/eprint/2329/5/BAB\_III.pdf.

[18] D. Lavarino and W. Yustanti, “RANCANG BANGUN E – VOTING BERBASIS WEBSITE DI UNIVERSITAS NEGERI SURABAYA,” *J. Manaj. Inform.*, vol. 6, pp. 72–81, 2016.

[19] T. Haryanto, “Pengenalan dan Sintaks Dasar CSS,” 2016. https://www.codepolitan.com/pengenalan-dan-sintaks-dasar-css (accessed Feb. 16, 2021).

[20] O. Pahlevi, A. Mulyani, and M. Khoir, “Sistem Informasi Inventori Barang Menggunakan Metode Object Oriented Di Pt. Livaza Teknologi Indonesia Jakarta,” *J. PROSISKO*, vol. 5, no. 1, 2018, [Online]. Available: https://livaza.com/.

[21] M. Clinic, “CT Scan.” https://www.mayoclinic.org/tests-procedures/ct-scan/about/pac-20393675 (accessed Nov. 15, 2020).

[22] Kemenkes RI, “QnA : Pertanyaan dan Jawaban Terkait COVID-19.” https://covid19.kemkes.go.id/qna-pertanyaan-dan-jawaban-terkait-covid-19/#Apakah\_Coronavirus\_dan\_COVID-19\_itu (accessed Nov. 15, 2020).

[23] World Health Organization, “Deteksi antigen dalam diagnosis infeksi SARS-CoV-2 menggunakan imunoasai cepat,” no. September, 2020, [Online]. Available: https://www.who.int/docs/default-source/searo/indonesia/covid19/deteksi-antigen-dalam-diagnosis-infeksi-sars-cov-2-menggunakan-imunoasai-cepat.pdf?sfvrsn=222f2be3\_2.

# LAMPIRAN

1. Contoh Lampiran 1

Ini adalah isi dari lampiran 1

1. Contoh Lampiran 2

Ini adalah isi dari lampiran 2

# BIODATA PENULIS

Biodata penulis berisi terkait dengan identitas penulis (nama, kontak email), riwayat akademis (pendidikan) penulis ditulis dari yang paling , skill, serta prestasi penulis.

A. Identitas

|  |  |  |
| --- | --- | --- |
| Nama | : |  |
| NIM | : |  |
| Tempat, tanggal lahir | : |  |
| Alamat | : |  |
| No. Telp. | : |  |
| Alamat e-mail | : |  |

B. Riwayat Pendidikan Akademik

| **Periode** | **Jenjang** | **Institusi** |
| --- | --- | --- |
| 2012 – 2016 | S1 | Teknik Elektro Universitas Jenderal Soedirman |
| 2009 – 2012 | SMA | SMAN xx Purwokerto |
| 2006 – 2009 | SMP | SMPN xx Sokaraja |

C. Riwayat Pendidikan Non Formal (jika ada)

| **Tahun** | **Keahlian** | **Penyelenggara** | **Kota** |
| --- | --- | --- | --- |
| 2014 | Bahasa Inggris Tingkat Mahir | Lembaga Kursus xxxxx | Purwokerto |
| 2013 | Kemanan Jaringan Mikrotik Tingkat Mahir | Lembaga xxxxxxx | Jakarta |

D. Prestasi

| **Tahun** | **Tingkat** | **Prestasi** |
| --- | --- | --- |
| 2014 | Nasional | Juara 1 lomba penulisan karya ilmiah, Yogyakarta |
| 2013 | Internasional | Medali emas olimpiade sains internasional, Dakka, India |

E. Keahlian (tuliskan secara diskriptif)

Memiliki minat di bidang pengembangan perangkat tertanam. Mampu merancang sistem embedded berbasiskan mikro kontroler atmega, arduino dan ESP8266. Terlibat secara aktif dalam kegiatan asistem Laboratorium Sistem Telekomunikasi dan Informasi sebagai asisten praktikum Algoritma dan Struktur Data, Jaringan Komputer, dan Dasar Pemrograman.

1. [↑](#footnote-ref-1)