

**PERBANDINGAN KINERJA ARSITEKTUR *DEEP LEARNING VGGNET*
DAN *RESNET* PADA IDENTIFIKASI KUPU-KUPU**

(Skripsi)

Oleh :

**ROCHMAT ALI
1617051094**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2023**

ABSTRAK

PERBANDINGAN KINERJA ARSITEKTUR DEEP LEARNING *VGGNET* DAN *RESNET* PADA IDENTIFIKASI KUPU-KUPU

Oleh

ROCHMAT ALI

Kupu-kupu merupakan makhluk hidup yang berpengaruh terhadap keseimbangan ekosistem. Kupu-kupu adalah salah satu serangga yang berperan menjadi perantara penyerbukan tanaman, sehingga kupu-kupu berperan dalam keseimbangan ekosistem. Identifikasi terhadap kupu-kupu diperlukan untuk membantu orang awam atau pengunjung taman kupu-kupu, dalam menambah pengetahuannya mengenai berbagai jenis kupu-kupu yang ada. Identifikasi di mudahkan dengan adanya teknologi *deep learning*, serta perkembangan algoritma *Convolutional Neural Network* (CNN) dan berbagai arsitektur yang di kebembangkan oleh algoritma tersebut. Arsitektur *VggNet* dan *ResNet* digunakan untuk membantu proses klasifikasi kelas kupu-kupu. Kedua arsitektur digunakan, karena dianggap sebagai arsitektur yang memiliki tingkat akurasi yang baik pada identifikasi kupu-kupu. Penelitian ini dilakukan menggunakan 800 gambar dari 8 kelas, masing-masing kelas terdiri dari 100 gambar. Percobaan dilakukan secara *online* dan *offline*. Hasil yang di berikan ialah perbandingan kedua arsitektur dari masing-masing metode. Akurasi terbesar di dapatkan oleh arsitektur *ResNet*, yaitu sebesar 99,37%.

Kata Kunci: Kupu-kupu, *deep learning*, *VggNet*, *ResNet*

ABSTRACT

COMPARISON OF PERFORMANCE BETWEEN DEEP LEARNING ARCHITECTURES VGGNET AND RESNET IN BUTTERFLY IDENTIFICATION

By

ROCHMAT ALI

Butterflies are living creatures that have an impact on ecosystem balance. Butterflies are one of the insects that play a role as intermediaries in plant pollination, thus contributing to ecosystem balance. Identification of butterflies is necessary to assist the general public or visitors of butterfly gardens in expanding their knowledge about the various species of butterflies. Identification is facilitated by the presence of deep learning technology, as well as the development of Convolutional Neural Network (CNN) algorithms and various architectures developed by these algorithms. VggNet and ResNet architectures are used to assist in the classification process of butterfly classes. Both architectures are used because they are considered to have a high level of accuracy in butterfly identification. This research was conducted using 800 images from 8 classes, with each class consisting of 100 images. Experiments were conducted online and offline. The results provided a comparison of the two architectures for each method. The highest accuracy was obtained by the ResNet architecture, which reached 99.37%.

Keywords: Butterflies, deep learning, VggNet, ResNet

**PERBANDINGAN KINERJA ARSITEKTUR *DEEP LEARNING VGGNET*
DAN *RESNET* PADA IDENTIFIKASI KUPU-KUPU**

Oleh :

ROCHMAT ALI

Skripsi

**Sebagai Salah Satu Syarat untuk Mencapai Gelar SARJANA
KOMPUTER**

Pada

**Jurusan Ilmu Komputer
Fakultas Matematika dan Ilmu Pengetahuan Alam**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2023**

Judul Skripsi : **PERBANDINGAN KINERJA ARSITEKTUR
DEEP LEARNING VGGNET DAN RESNET
PADA IDENTIFIKASI KUPU-KUPU**

Nama Mahasiswa : **Rochmat Afi**

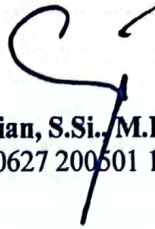
Nomor Pokok Mahasiswa : 1617051094

Program Studi : Ilmu Komputer

Fakultas : Matematika dan Ilmu Pengetahuan Alam

MENYETUJUI

1. **Komisi Pembimbing**



Rico Andrian, S.Si., M.Kom.
NIP. 19750627 200501 1 001

2. **Ketua Jurusan Ilmu Komputer**



Didik Kurniawan, S.Si., M.T.
NIP. 19800419 200501 1 004

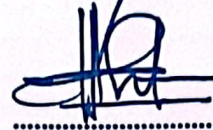
MENGESAHKAN

1. Tim Penguji

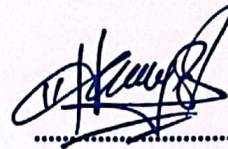
Ketua : Rico Andrian, S.Si., M.Kom.

.....


**Penguji
Bukan Pembimbing : Dr. rer. nat. Akmal Junaidi, M.Sc.**



**Penguji
Bukan Pembimbing : Yunda Henigtyas, M.Kom.**

.....


2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



Dr. Eng. Heri Satria, S.Si., M.Si.
NIP-19711001 2005 01 1 002

Tanggal Lulus Ujian Skripsi : 20 Juni 2023

PERNYATAAN

Saya yang bertanda tangan di bawah ini :

Nama : Rochmat Ali

Npm : 1617051094

Dengan ini menyatakan bahwa skripsi saya yang berjudul “Perbandingan Kinerja Arsitektur *Deep Learning VggNet Dan ResNet* pada Identifikasi Kupu-Kupu” merupakan karya saya sendiri dan bukan karya orang lain. Semua tulisan yang tertuang dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila di kemudian hari terbukti skripsi saya merupakan hasil penjiplakan atau dibuat orang lain, maka bersedia menerima sanksi berupa pencabutan gelar yang telah saya terima.

Bandar Lampung, 20 Juni 2023



Rochmat Ali

NPM. 1617051094

RIWAYAT HIDUP



Penulis lahir di Bandar Jaya, 24 Juni 1998, dilahirkan sebagai anak pertama dari dua bersaudara dari pasangan Bapak Santoni Ali dan Ibu Rosada. Pendidikan yang telah ditempuh oleh penulis diantaranya, menyelesaikan pendidikan taman kanak-kanak di TK Islam Terpadu Bustanul Ulum, Lampung Tengah pada tahun 2004. Penulis menyelesaikan pendidikan dasar di SD Islam Terpadu Bustanul Ulum Lampung Tengah pada tahun 2010. Penulis menyelesaikan pendidikan menengah dalam program kelas Akselerasi di SMP N 3 Way Pengubuan Lampung Tengah pada tahun 2012. Penulis melanjutkan pendidikan menengah atas di SMK N 1 Terusan Nunyai Lampung Tengah dan lulus pada tahun 2015. Perjalanan pendidikan penulis dilanjutkan dengan terdaftar sebagai mahasiswa Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung melalui jalur Seleksi Bersama Masuk Perguruan Tinggi (SBMPTN) pada tahun 2016.

Selama menjadi mahasiswa, penulis aktif mengikuti beberapa kegiatan antara lain:

1. Anggota Abacus Himpunan Mahasiswa Ilmu Komputer periode 2016/2017.
2. Pada bulan Januari 2020 penulis melaksanakan Kuliah Kerja Nyata (KKN) di Desa Rantau Fajar, Raman Utara, Lampung Timur.
3. Melaksanakan Kerja Praktik pada bulan Juni periode 2019/2020 di Jurusan Ilmu Komputer.

4. Pada bulan Oktober 2020 penulis mengikuti ujian sertifikasi yang diselenggarakan oleh BNSP dengan kualifikasi atau kompetensi sebagai *Junior Web Programming*.

PERSEMBAHAN

Alhamdulillahillobbilamin

Puji dan syukur atas kehadiran Allah SWT atas segala Rahmat dan Karunia-Nya sehingga saya dapat menyelesaikan skripsi ini. Shalawat serta salam selalu tersampaikan kepada Nabi Muhammad SAW.

Kupersembahkan karya ini kepada:

Kedua Orang Tuaku

Yang selalu senantiasa memberikan segala yang terbaik untukku, selalu mendukung dan mendoakanku tanpa henti. Terimakasih atas kasih sayang, pengorbanan dan kerja keras kalian yang sampai kapanpun tak akan pernah mampu terbalaskan. Terimakasih selalu ada untukku dan telah menjadi orangtua sempurna untukku.

Adikku Tercinta

Yang telah memberikan semangat dan dukungannya selama ini, semoga kelak kita menjadi anak yang membanggakan kedua orangtua, karena kita berdua yang akan meneruskan perjuangan keluarga.

Teman-Teman Jurusan Ilmu Komputer 2016

Yang selalu memberikan semangat dan dukungan.

Almamater Tercinta, Universitas Lampung dan Jurusan Ilmu Komputer

Tempat bernaung mengemban semua ilmu untuk menjadi bekal hidup.

SANWACANA

Segala Puji bagi Allah, atas limpahan nikmat-Nya yang diberikan kepada penulis sehingga dapat menyelesaikan Skripsi berjudul “Perbandingan Kinerja Arsitektur *Deep Learning VggNet* Dan *ResNet* Pada Identifikasi Kupu-Kupu”. Tidak lupa shalawat dan salam senantiasa dicurahkan kepada Nabi Muhammad SAW, suri teladan yang mampu membuka sesuatu yang terkunci, penutup dari semua yang terdahulu, penolong kebenaran dengan jalan yang benar, dan petunjuk kepada jalan-Mu yang lurus.

Pada kesempatan ini penulis mengucapkan terimakasih kepada semua pihak yang telah membantu dan memiliki peran besar dalam penyusunan skripsi ini, yaitu :

1. Allah SWT, yang telah memberikan nikmat, rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan Skripsi ini dengan selamat hingga akhir.
2. Kedua orangtua yang selalu memberikan doa, motivasi, semangat, dan bantuan yang begitu melimpah dan tidak akan pernah bisa terhitung nilainya.
3. Adikku yang selalu memberikan dukungan sehingga penulis terus semangat untuk memotivasi diri menjadi contoh kakak yang terus lebih baik.
4. Bapak Didik Kurniawan, S.Si., M.T. selaku Kepala Jurusan Ilmu komputer Universitas Lampung.
5. Bapak Rico Andrian, S.Si., M.Kom selaku dosen pembimbing atas kesediannya dan kesabarannya untuk memberikan bimbingan, motivasi, dan arahan kepada penulis dalam proses penyelesaian skripsi.
6. Ibu Yunda Heningtyas, M.Kom. selaku dosen pembahas yang telah memberikan kritik, saran serta masukan kepada penulis.
7. Bapak Dr.rer.nat. Akmal Junaidi, M.Sc. selaku Dosen pembahas yang telah memberikan kritik, saran serta masukan kepada penulis.

8. Bapak dan Ibu Dosen Jurusan Ilmu Komputer Universitas Lampung yang telah banyak memberikan ilmu dan pengetahuan hidup selama penulis menjadi mahasiswa.
9. Seluruh Staff Administrasi Jurusan Ilmu Komputer yang selama ini selalu membantu penulis dalam hal pemberkasan dan informasi selama perkuliahan
10. Teman-Teman seperjuangan, yang telah banyak membantu penulis selama proses bimbingan skripsi berjalan.
11. Teman-teman kosan, yang selalu menghibur dan memberikan semangat selama ini kepada penulis.
12. Teman-teman mahasiswa Ilmu Komputer angkatan 2016 yang telah memberikan dukungan yang tiada henti sera terimakasih telah memeberikan warna dalam kehidupan perkuliahan penulis.
13. Teman-teman seperjuangan di grup “wisuda semester ini” yang telah memberi informasi-informasi tentang segala hal.
14. Fika Merliana terimakasih telah memberi banyak sekali dukungan dan *support* penuh kepada penulis dalam segala hal.
15. Eldam, Rendi, Ilham, terimakasih sudah menemani selama beberapa tahun ngegabut *see you on top*.
16. Semua pihak yang turut membantu penulis dalam penyusunan laporan skripsi ini.

Penulis menyadari bahwa masih banyak kekurangan dalam penulisan skripsi ini, akan tetapi semoga skripsi ini dapat membawa manfaat dan keberkahan bagi perkembangan ilmu pengetahuan terutama bagi semua civitis Ilmu Komputer Universitas Lampung.

□□□□□□□□

□□□□

□□□Bandar □□ampung, 20 Juni 2023

Rochmat Ali
NPM. 1617051094

DAFTAR ISI

	Halaman
DAFTAR ISI.....	ii
DAFTAR TABEL	iii
DAFTAR GAMBAR.....	iv
I. PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	4
1.3 Batasan Masalah.....	4
1.4 Tujuan Penelitian.....	5
1.5 Manfaat Penelitian.....	5
II. TINJAUAN PUSTAKA.....	6
2.1 Kupu-Kupu	6
2.2 <i>Deep Learning</i>	9
2.3 <i>Convolutional Neural Network</i>	9
2.4 <i>Arsitektur Convolutional Neural Network</i>	10
2.5 <i>Residual Network (ResNet)</i>	13
2.6 <i>Visual Geometry Group Network (VggNet)</i>	15
2.7 Perhitungan Akurasi	17
2.8 <i>Confusion Matrix</i>	17
III. METODOLOGI PENELITIAN	20
3.1 Tempat dan Waktu Penelitian	20
3.3.1 Tempat Penelitian.....	20
3.3.2 Waktu Penelitian	20
3.2 Alat dan Bahan	20
3.3.1 Perangkat Keras (<i>Hardware</i>).....	21
3.3.2 Perangkat Lunak (<i>Software</i>)	21
3.3.3 Bahan Penelitian.....	21

3.3 Tahap Penelitian	24
3.3.1 <i>Dataset</i>	25
3.3.1 <i>Preprocessing</i>	25
3.3.2 Pemisahan Data	26
3.3.3 Tahap Pelatihan	26
3.3.4 Pengujian Model	27
3.3.5 Hasil Pengujian	27
IV. HASIL DAN PEMBAHASAN	34
4.1 Hasil Klasifikasi Arsitektur <i>ResNet</i>	34
4.2 Hasil Klasifikasi Arsitektur <i>VggNet</i>	38
4.3 Pembahasan	41
V. SIMPULAN DAN SARAN	49
5.1 Simpulan	49
5.2 Saran	50
DAFTAR PUSTAKA	51
LAMPIRAN	53

DAFTAR TABEL

Tabel	Halaman
Tabel. 1. Lapisan <i>Vgg19</i>	16
Tabel. 2. <i>Confusion Matrix</i>	18
Tabel. 3. Waktu Penelitian	20
Tabel. 4. Jumlah <i>Dataset</i>	25
Tabel. 5. Model Parameter	27
Tabel. 6. <i>Recall, precision, f1 score ResNet50.</i>	43
Tabel. 7. <i>Recall, precision, f1 score Vgg16</i>	46

DAFTAR GAMBAR

Gambar	Halaman
1. Tubuh Kupu-Kupu	6
2. Dimensi CNN.....	10
3. Arsitektur CNN.....	10
4. Operasi Konvolusi.....	11
5. Ilustrasi <i>Dropout</i>	12
6. Model Arsitektur <i>ResNet</i>	13
7. Blok <i>ResNet</i>	14
8. Arsitektur <i>Vgg16</i>	15
9. Kupu-kupu Jarak	21
10. Kupu-kupu Batik Cap	22
11. Kupu-kupu Hijau Biru.....	22
12. Kupu-Kupu Pantat Merah	23
13. Kupu-kupu Raja Limau.....	23
14. Kupu-kupu Jojo.....	23
15. Kupu-kupu Harimau Kuning Hijau.....	24
16. Kupu-kupu Raja Helena.....	24
17. Tahap Penelitian.....	24
18. Hasil <i>Scaling</i>	25
19. <i>Hyperparameter ResNet50</i>	30
20. Model <i>ResNet50</i>	30
21. Grafik Perbandingan akurasi model <i>ResNet50</i>	37
22. perbandingan <i>Google Colab</i> dan <i>Jupyter Notebook ResNet50</i>	37
23. <i>Hyperparameter Vgg16</i>	39
24. Model <i>Vgg16</i>	39
25. Grafik Perbandingan akurasi model <i>Vgg16</i>	40
26. Perbandingan <i>Google Colab</i> dan <i>Jupyter Notebook Vgg16</i>	41

27. Hasil <i>confusion matrix ResNet50</i>	42
28. Perbandingan Kupu Raja Limau dan Kupu Hijau Biru (<i>ResNet50</i>)	44
29. Hasil <i>confusion matrix Vgg16</i>	45
30. Perbandingan Perbandingan Kupu Jarak dan Kupu Batik Cap.....	47
31. Perbandingan Kupu Jojo dan Kupu Pantat Merah	48
32. Perbandingan Kupu Raja Limau dan Kupu Hijau Biru (<i>Vgg16</i>)	48

I. PENDAHULUAN

1.1 Latar Belakang

Kupu-kupu adalah kelompok serangga yang termasuk ke dalam ordo *Lepidoptera*, yang berarti memiliki sayap yang bersisik. Kupu-kupu merupakan jenis serangga yang paling banyak dikenal karena bentuk dan warna yang indah serta coraknya yang beragam sehingga banyak menarik perhatian, dan sering dijumpai pada siang hari. Kupu-kupu juga merupakan salah satu serangga yang berperan menjadi perantara penyerbukan tanaman secara biotik, yang artinya kupu-kupu dapat memindahkan serbuk sari dari *anther* ke *stigma* sehingga secara ekologis kupu-kupu berperan dalam mempertahankan keseimbangan ekosistem dan memperkaya keragaman hayati. Indikator kualitas lingkungan juga dapat dilihat dari keberadaan kupu-kupu yang tinggal di lingkungan tersebut. Kupu-kupu yang berada di suatu area dapat memberikan indikasi bahwa daerah tersebut masih alami dan belum terganggu, begitu pula sebaliknya jika di suatu area tidak terdapat kupu-kupu atau menurunnya jumlah spesies kupu-kupu di area tersebut maka menandakan bahwa daerah tersebut sudah terganggu dan mengalami penurunan kualitas lingkungan.

Lampung merupakan satu-satunya kota yang memiliki penangkaran kupu-kupu yang dilakukan di habitat aslinya (*In Situ*), yaitu Taman Kupu-kupu Gita Persada yang berada di Jalan Wan Abdurrahman, Hutan, Kecamatan Hutan, Lampung, Indonesia. Taman Kupu-Kupu Gita Persada memiliki banyak sekali spesies kupu-kupu yang dikembangkan, namun pada dasarnya Taman Kupu-Kupu Gita Persada masih membutuhkan identifikasi kupu-kupu yang bertujuan untuk membedakan spesies kupu-kupu yang ada di Taman Kupu-Kupu Gita Persada. Mata manusia dapat membedakan spesies kupu-kupu secara langsung, namun masih tergolong sulit karena mata

manusia memiliki keterbatasan untuk membedakan tekstur dan warna yang khas pada spesies kupu-kupu, sehingga dibutuhkan teknologi untuk mempermudah mengenali jenis kupu-kupu dengan ciri-ciri dan pola yang terdapat pada kupu-kupu. Teknologi *deep learning* dapat digunakan sebagai teknologi dalam proses identifikasi kupu-kupu.

Deep learning merupakan model jaringan syaraf tiruan yang telah menyebar dan berkembang seiring dengan perkembangan teknologi dalam pengenalan citra. *Deep Learning* dan kecerdasan buatan termasuk dalam bidang *computer vision* dalam mengekstraksi informasi dari gambar. *Computer vision* mengadaptasi kemampuan penglihatan manusia dalam mengambil informasi dari objek gambar (Borugadda, *et al.*, 2021). *Deep learning* dapat memiliki akurasi yang tinggi, salah satu metode *deep learning* yang paling sering digunakan adalah *Convolutional Neural Network (CNN)*. *Convolutional Neural Network (CNN)* merupakan salah satu algoritma *deep learning* yang merupakan evolusi dari *Multilayer Perceptron (MLP)* yang didesain untuk mengolah data secara dua dimensi. Prinsip pengoperasian CNN adalah meniru aktivitas otak manusia untuk mengenali objek yang dilihatnya. Komputer sekarang dapat melihat dan membedakan objek yang berbeda. Fitur ini disebut *image recognition*. *Neuron- neuron* yang dimiliki oleh CNN disusun secara tiga dimensi (memiliki panjang, lebar,tinggi), sehingga CNN sangat efektif dan efisien untuk menganalisis *image/gambar* (Primartha, 2020). CNN sendiri memiliki beberapa arsitektur diantaranya *VggNet* dan *ResNet*. *Visual Geometry Group Network* atau yang lebih dikenal dengan *VggNet* merupakan lapisan CNN yang disusun oleh Karen Simonyan dan Andrew Zisserman dari *University of Oxford* yang telah merilis serangkaian model jaringan konvolusi yang dimulai dengan *visual geometry group* yang dapat diterapkan pada pengenalan wajah dan klasifikasi gambar, terdapat 2 jenis *VggNet* yaitu *Vgg16* dan *Vgg19*, meskipun berbeda akan tetapi kedua *VggNet* tersebut memiliki tujuan yang sama, yaitu untuk memahami bagaimana kedalaman jaringan konvolusi mempengaruhi tingkat akurasi klasifikasi serta pengenalan gambar skala besar.

Penelitian tentang identifikasi kupu-kupu menggunakan arsitektur CNN telah dilakukan oleh beberapa peneliti sebelumnya. Pengujian yang dilakukan yaitu melakukan penelitian terhadap kupu-kupu yang ada di *website of Butterflies Monitoring & Photography Society of Turkey* dengan *dataset* yang digunakan pada penelitian ini dibagi menjadi *data training* sebesar 80% dan *data testing* sebesar 20%. *Dataset* berjumlah 17.769 citra kupu-kupu dan memiliki 10 kelas kupu-kupu. Arsitektur yang digunakan pada penelitian ini adalah *VggNet* dan *ResNet* dan didapatkan hasil akurasi pada masing-masing arsitektur sebesar 79,49% dan 70,16%. (Almryad & Kutucu, 2020).

Penelitian lain yang juga menggunakan arsitektur *VggNet* dilakukan untuk mengidentifikasi penyakit. Pengujian yang dilakukan membahas mengenai mendeteksi penyakit *Alzheimer* menggunakan klasifikasi gambar otak. Penelitian ini menggunakan metode klasifikasi CNN dengan arsitektur *VggNet* dan didapatkan hasil akurasi sebesar 95,73% untuk *set* validasi. (Jain, *et al.*, 2018)

Arsitektur lain yang dapat digunakan untuk identifikasi kupu-kupu adalah *Residual Networks (ResNet)*, yaitu sebuah model arsitektur CNN yang memenangkan ILSVRC pada tahun 2015 dengan menggunakan *skip connection* dan *Features batch normalization*. *ResNet* memungkinkan pelatihan neural network dengan 152 layer dan mampu mengurangi kompleksitas. Tingkat kesalahan yang dicapai dari arsitektur ini adalah sebesar 3.57% pada *Top-5*, sehingga mampu mengalahkan kinerja *human-level* pada *dataset* yang diberikan (He *et al.*, 2016).

Penelitian terdahulu terkait identifikasi kupu-kupu menggunakan *ResNet* sebagai arsitekturnya. Penelitian tersebut menggunakan *dataset* kupu-kupu baru dengan 34.024 gambar milik 315 spesies kupu-kupu dari India. model arsitektur *ResNet*. Hasil klasifikasi model dan teknik yang diusulkan

mencapai akurasi *top-1* (94,44%), akurasi *top-3* (98,46%) dan akurasi *top-5* (99,09%) menggunakan model *ResNet-152* (Hari Theivapraksham, 2020).

Arsitektur *ResNet* tidak hanya mampu mengklasifikasikan kupu-kupu, namun dapat digunakan untuk klasifikasi dan deteksi gambar lainnya seperti yang dilakukan penelitian terdahulu, penelitian tersebut membahas mengenai mendeteksi dan mengklasifikasikan *image* dari *CT image* patah tulang. Hasil dari penelitian menyatakan bahwa kelayakan untuk menemukan daerah yang retak menggunakan CNN dengan *ResNet* sebagai model yang telah dilatih sebelumnya. *ResNet* menunjukkan akurasi 97,89% dan *runtime* 5 menit dalam mengklasifikasikan patahan. (Pranata *et al.*, 2019).

Penelitian yang akan dilakukan yaitu membandingkan kinerja arsitektur *deep learning VggNet* dan *ResNet* pada identifikasi kupu-kupu di Taman Kupu-Kupu Gita Persada. Penelitian ini dapat mengetahui perbandingan tingkat akurasi dari arsitektur terhadap citra kupu-kupu.

1.2 Rumusan Masalah

Rumusan masalah dalam penelitian ini adalah bagaimana perbandingan tingkat akurasi terhadap identifikasi kupu-kupu menggunakan arsitektur *VggNet* dan *ResNet*.

1.3 Batasan Masalah

Batasan masalah pada penelitian ini adalah sebagai berikut:

- a. Citra kupu-kupu yang digunakan merupakan sisi sayap bagian atas.
- b. Spesies kupu-kupu yang akan diidentifikasi adalah sebanyak 8 spesies diantaranya, *Ariadne ariadne*, *Cethosia penthesilea*, *Papilio peranthus*, *Pachliopta aristolochiae*, *Papilio memnon*, *Papilio nephelus*, *Parantica aspasia*, *Troides helena*.
- c. *Dataset* yang digunakan berjumlah 800 citra, dengan masing- masing spesies berjumlah 100 citra kupu-kupu.

- d. *Dataset* yang digunakan merupakan kupu-kupu yang ada di Taman Kupu- kupu Gita Persada.

1.4 Tujuan Penelitian

Tujuan dari penelitian adalah membandingkan kinerja arsitektur *deep learning VggNet* dan *ResNet* dalam identifikasi kupu-kupu.

1.5 Manfaat Penelitian

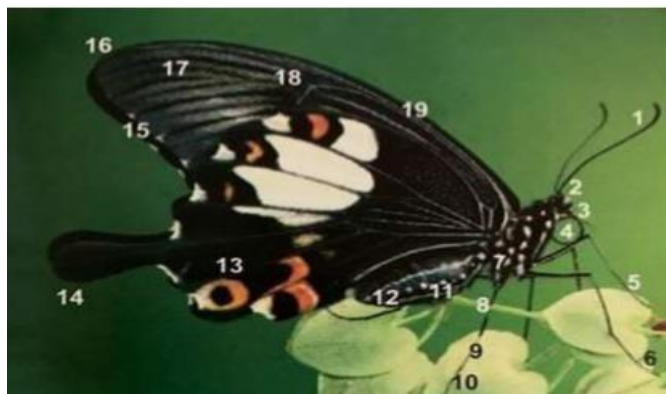
Manfaat dari penelitian ini adalah sebagai berikut:

- a. Mengetahui perbandingan kinerja arsitektur *deep learning VggNet* dan *ResNet* dalam identifikasi kupu-kupu dengan memberikan pemahaman tentang kelebihan dan kekurangan yang dimiliki oleh masing-masing arsitektur.
- b. Penelitian ini dapat memberikan kemudahan proses identifikasi kupu-kupu terhadap pihak pengelola Taman Kupu- Kupu Gita Persada dengan menggunakan Teknologi *deep learning*.
- c. Penelitian ini dapat membantu para *developer* untuk menjadi bahan pengembangan aplikasi identifikasi kupu-kupu menggunakan *computer vision*.
- d. Penelitian ini dapat dijadikan bahan rujukan penelitian lain mengenai identifikasi kupu-kupu.

II. TINJAUAN PUSTAKA

2.1 Kupu-Kupu

Kupu-kupu termasuk ke dalam ordo *Lepidoptera* yang memiliki kombinasi corak warna yang variatif. Kupu-kupu merupakan bagian kehidupan di alam, yaitu salah satu satwa penyerbuk pada pembuahan pada tumbuhan berbunga dan secara ekologis dapat mempertahankan keseimbangan ekosistem dan memperkaya keanekaragaman hayati (Koneri & Saroyo, 2012). Tubuh kupu-kupu terbagi menjadi 3 bagian yaitu Kepala, Thorak, dan Abdomen. Bagian kepala kupu-kupu memiliki sepasang antena dan satu probosis yang merupakan tabung penghisap panjang. Bagian thorak pada kupu-kupu terdapat 2 pasang sayap serta 3 pasang kaki. Sayap kupu-kupu terdapat membran dengan venasi yang digunakan untuk klasifikasi kupu-kupu. Venasi mempunyai nama atau sebuah kode yang berbeda pada setiap kupu-kupu untuk mempermudah dalam membedakan pola bentuk dan warna yang terdapat pada kupu-kupu (Soekardi, *et al.*, 2016).



Gambar 1. Tubuh Kupu-Kupu (Soekardi, *et al.*, 2016).

Bagian pada Gambar 1. Tubuh Kupu-Kupu (Soekardi, *et al.*, 2016). yaitu :

- | | | |
|------------------|---------------------------|--------------------|
| 1.) Antena | 8.) Femur | 15.) Margin Luar |
| 2.) Kepala | 9.) Tibia Kaki Belakang | 16.) Apex |
| 3.) Mata Majemuk | 10.) Tarsus Kaki Belakang | 17.) Vena |
| 4.) Probosis | 11.) Segmen Abdominal | 18.) Sayap Depan |
| 5.) Kaki Depan | 12.) Abdomen | 19.) Margin Costal |
| 6.) Kaki Tengah | 13.) Sayap Belakang | |
| 7.) Thoraks | 14.) Ekor Sayap | |

Kupu-kupu adalah serangga yang mengalami metamorfosis sempurna (*Holometabola*), karena kupu-kupu biasanya memiliki bentuk pra-dewasa (*larva* dan *Pupa*) yang sangat berbeda dengan bentuk dewasa (*Imago*). Kupu-kupu mengalami empat fase selama hidupnya, yaitu fase telur, *larva*, *Pupa* dan *Imago* (Jumuar, 2002).

a. Telur

Telur kupu-kupu berukuran kecil, beragam bentuknya tergantung pada jenisnya ada yang memanjang, *oval*, bulat, berbentuk botol dan keriput. Warna telur kupu-kupu pun beragam, cangkang telur ada yang halus, ada pula yang seperti terpahat. Bagian bawah telur selalu rata. Bagian atas telur terdapat *micropile*, yaitu lubang kecil tempat masuknya *spermatozoid*. Fase telur rata-rata berkisar antara 4-10 hari (Amir *et al.*, 2008).

b. *Larva*

Larva adalah fase yang sangat aktif melakukan aktivitas makan yang diperlukan *larva* untuk tumbuh dan berkembang. Selama stadium *larva*, umumnya kupu-kupu mengalami lima kali pergantian kulit kitin (*molting*). Banyaknya frekuensi *molting* berbeda-beda, tergantung pada jenis kupu-kupu dan kondisi kesehatan *larva*-nya. Badan *larva* terdiri dari tiga bagian, yaitu bagian kepala, torak dan abdomen. Kapsul kepala pada *larva* mengalami *sklerotisasi*, sehingga kitin pada bagian kapsul

kepala ini lebih keras dan kuat dibandingkan dengan kulit kitin pada torak dan abdomen. Kepala larva mempunyai sepasang mata yang terdiri dari enam stemata (*ocelli*), dan sepasang antena dengan tiga segmen yang tidak berkembang, *setae*, *labrum*, *maksilla*, *spinneret* dan mulut. Mulut *larva* bertipe penggigit dan pengunyah, makanan larva yakni dedaunan (Jumuar, 2002).

c. *Pupa*

Pupa adalah fase ketika *larva* istirahat. *Pupa* merupakan bentuk peralihan yang dicirikan dengan terjadinya perombakan dan penyusunan kembali alat-alat tubuh bagian dalam. Fase *Pupa* juga merupakan masa persiapan sebelum terjadi pergantian kulit yang tetap pada fase *Imago*. *Larva* yang akan mengalami proses metamorfosis dari bentuk *larva* menjadi bentuk *Pupa* (*Pupasi*), terlebih dahulu akan mengalami proses pra-*Pupa*. Pada tahap Pra-*Pupa* aktivitas *larva* akan berkurang seakan-akan lemas atau tertidur, dan *larva* sudah tidak melakukan aktivitas makan lagi. Kejadian ini akan berlangsung selama 2-3 hari sampai *larva* selesai melakukan *Pupasi* dan menjadi *Pupa* (Achmad, 2017).

d. *Imago*

Imago adalah fase dewasa dari *Lepidoptera*, dan merupakan fase berkembangbiak. Seperti serangga lainnya, badan kupu-kupu dibedakan menjadi kepala, torak dan abdomen. Kepala kecil, *hypognathous*. Antena satu pasang, panjang dan ramping, terdiri dari banyak segmen. Mata majemuk satu pasang, besar. Mata *ocelus* dua buah, tersembunyi di bawah sisik-sisik di kepala. Alat mulut disesuaikan untuk mengisap (*siphoning type mouthpart*), *labrum* mereduksi; *maksila* membentuk satu pasang *probosis* panjang yang saling melekat, digulung pada waktu tidak dipergunakan; *mandibula* kecil dan mereduksi; *labium* mereduksi, tetapi *palpus labialis* berkembang untuk menjalankan fungsi tertentu dalam memilih makanannya (Amir *et al.*, 2008).

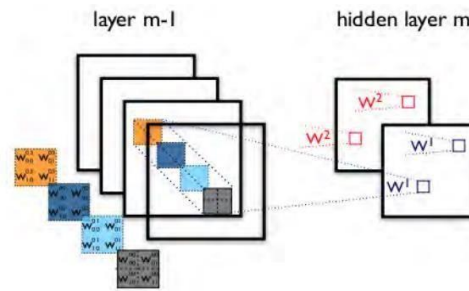
2.2 *Deep Learning*

Deep Learning merupakan cabang *Machine Learning* yang terinspirasi dari korteks otak manusia, menggunakan jaringan syaraf tiruan yang memiliki banyak *hidden layer*. Proses *Deep Learning*, sebuah komputer belajar mengklasifikasi secara langsung dari gambar atau suara (Batubara & Awangga, 2020). Model *Deep Learning* dilatih menggunakan kumpulan data besar dan melintasi beberapa lapisan untuk membaca, menyimpan, memproses, dan mengklasifikasikan *input*, sehingga *Deep Learning* seringkali mengungguli kinerja manusia. Metode *Deep Learning* menggunakan CPU dan RAM dalam proses komputasi, dan juga memanfaatkan GPU sehingga proses komputasi berlangsung lebih cepat (Ilahiyah & Nilogiri, 2018).

2.3 *Convolutional Neural Network*

Convolutional Neural Network (CNN) adalah model *Deep Learning* yang digunakan untuk klasifikasi citra, deteksi kemiripan dan pengenalan objek (J. wang *et al*, 2016). CNN juga merupakan pengembangan dari *Multilayer Perceptron* (MLP) yang didesain untuk mengolah data dua dimensi dalam bentuk citra. CNN salah satu arsitektur *Deep Learning* yang paling efisien karena mampu menurunkan *error* atau galat sampai dengan 15,3%. Dua aspek utama CNN adalah mendeteksi tambahan fitur terlepas dari lokasi dalam *matrix input* (mis. Invarian lokasi), dan menyusun tambahan fitur menjadi representasi tingkat yang lebih tinggi. Konsep kerja CNN mempunyai kesamaan seperti MLP, namun setiap *neuron* pada CNN direpresentasikan dalam bentuk dua dimensi seperti citra dan suara, sedangkan pada MLP hanya dalam bentuk satu dimensi, CNN operasi *linier* menggunakan operasi konvolusi, sedangkan bobot memiliki bentuk empat dimensi yang merupakan *kernel* konvolusi (Primartha, 2020), dapat dilihat pada Gambar 2.

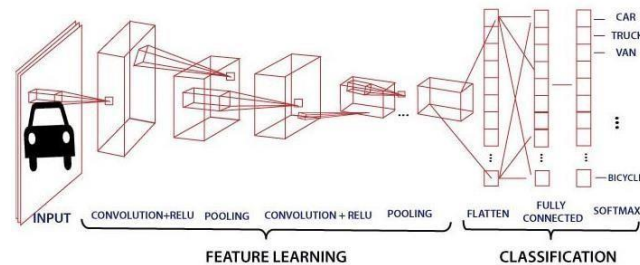
Neuron input x neuron output x tinggi x lebar



Gambar 2. Dimensi CNN (Suartika *et al.*, 2016)

2.4 Arsitektur *Convolutional Neural Network*

Arsitektur CNN memiliki komponen-komponen yang dapat dilihat pada Gambar 3.



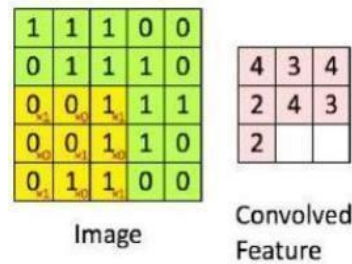
Gambar 3. Arsitektur CNN (Sena, 2017).

CNN terdiri dari beberapa *layer* seperti, *input layer*, *convolution layer*, *activation layer*, *Softmax*, *pooling layer*, *dropout*, dan *fully connected layer*. Lapisan-lapisan tersebut dapat dijelaskan sebagai berikut (Primartha, 2020).

a. *Convolutional Layer*

Output dari *layer* sebelumnya dilakukan oleh *convolution layer* dengan operasi konvolusi. *Layer* inilah yang menjadi dasar dari operasi CNN. Konvolusi adalah suatu istilah matematis yang berarti mengaplikasikan sebuah fungsi pada *output* fungsi lain secara berulang. konvolusi berarti mengaplikasikan sebuah *kernel* (kotak kuning) pada citra di semua *offset* yang memungkinkan. Kotak hijau secara keseluruhan adalah citra yang akan di konvolusi (Primartha, 2020), sehingga hasil konvolusi dari citra

tersebut dapat dilihat pada gambar di sebelah kanan, seperti yang ditunjukkan pada Gambar 4.



Gambar 4. Operasi Konvolusi (Suartika *et al.*, 2016)

b. *Activation Layer*

Activation Layer berfungsi untuk mengenalkan properti *non-linear* pada jaringan yang diikuti oleh lap konvolusi. Tujuan utama dari *activation layer* adalah untuk mengubah sinyal *input* dari sebuah *node* menjadi sinyal *output*. Hasil dari *output* akan dilanjutkan pada *layer* berikutnya. Fungsi aktivasi *Rectified Linear Unit* (ReLU) sebagian besar diterapkan dalam praktik kehidupan nyata karena efisien secara komputasi dan menyatu jauh lebih cepat daripada kebanyakan fungsi aktivasi lainnya. Fungsi berikut menyatakan *non-linear* ReLU mengambil *input* x pada persamaan (1) berikut ini:

$$f(x) = \max(0, x) \dots \dots \dots (1)$$

c. *Softmax*

Softmax merupakan fungsi aktivasi yang digunakan pada *layer output*. *Layer output* ini berada di bagian *fully connected layer*, yang membedakan kedua layer ini adalah penggunaan fungsi aktivasi *Softmax* pada *layer output* dan fungsi aktivasi ReLU pada *fully connected layer*. Persamaan fungsi *Softmax* ditunjukkan pada persamaan (2).

$$P_i = \frac{e^{x_i + \log(C)}}{\sum_{k=1}^n e^{x_k + \log(C)}} \dots \dots \dots (2)$$

Persamaan (2) x_i merupakan nilai *strength neuron* dan $\log(C)$ merupakan nilai konstan yang dapat ditentukan, namun secara

umum nilai yang ditetapkan adalah $\max(x)$. Fungsi aktivasi *softmax* memiliki ∂p_i turunan fungsi yang dijelaskan pada Persamaan (3).

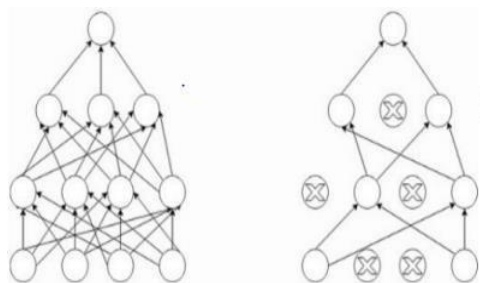
$$\begin{aligned} \frac{\partial p_i}{\partial x_i} &= (\delta_{ij} - p_j) \text{ dengan } \delta_{ij} \dots\dots\dots(3) \\ &= \begin{cases} 1, & \text{jika } i = j \\ 0, & \text{jika } i \neq j \end{cases} \end{aligned}$$

d. *Pooling Layer*

Pooling layer umumnya diperkenalkan antara dua *layer* konvolusi berturut-turut untuk mengurangi jumlah parameter dan perhitungan yang diperlukan oleh jaringan, sehingga membantu untuk mengatasi masalah *overfitting* sambil melatih *dataset*. *Max Pooling* adalah teknik *pooling* yang sangat umum untuk digunakan karena menghasilkan hasil yang lebih baik dalam aplikasi kehidupan nyata daripada teknik *pooling* lainnya.

e. *Dropout*

Dropout adalah teknik untuk mencegah masalah *overfitting*. *Dropout* direpresentasikan pada *fully connected layer* dengan cara kerja menonaktifkan beberapa *neuron* secara acak yang tidak diperlukan. Gambar 5. menunjukkan ilustrasi *dropout*.



Gambar 5. Ilustrasi *Dropout*. (Rahman *et al.*, 2018)

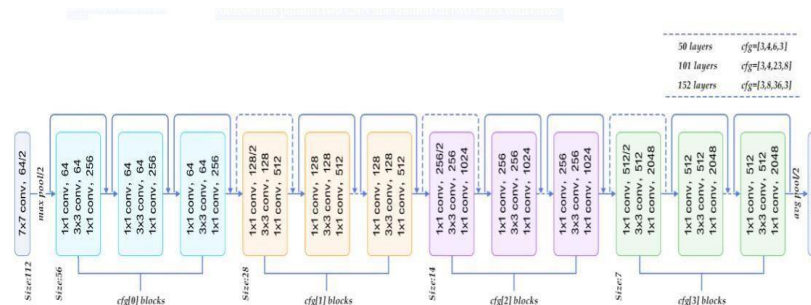
f. *Fully Connected Layer*

Fully connected layer memiliki koneksi penuh ke semua lapisan aktivasi *neuron* pada layer sebelumnya. *Layer* yang terhubung terakhir sepenuhnya dari jaringan menghasilkan *output* bersih dengan fungsi

aktivasi yaitu fungsi *softmax*, tergantung pada jumlah kelas atau kategori dalam jumlah klasifikasi.

2.5 Residual Network (ResNet)

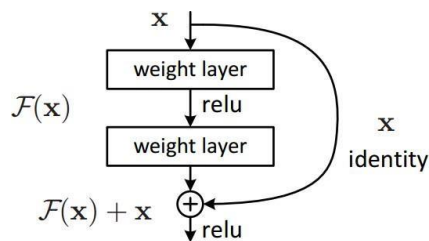
Residual Networks atau yang biasa disebut dengan *ResNet* adalah jaringan saraf klasik. Model ini adalah pemenang tantangan *ImageNet* pada tahun 2015. Arsitektur ini cukup *revolutioner* pada saat itu karena arsitektur ini menjadi *state of the art* pada saat itu tidak hanya dalam klasifikasi, namun dalam semua kategori termasuk *object detection*, dan *semantic segmentation*. *ResNet* memiliki tingkat akurasi yang baik karena memiliki *residual block* yang berfungsi untuk mencegah *vanishing gradient problem* yang mengakibatkan nilai *gradient* yang dicari menjadi kecil sehingga menjadi tidak akurat (He *et al.*, 2016). Gambar model arsitektur *ResNet* dapat dilihat pada Gambar 6.



Gambar 6. Model Arsitektur *ResNet* (He *et al.*, 2016)

Jaringan *ResNet* 50, 101, dan 152 layer, terdapat modifikasi pada *building block*. Modifikasi tersebut pada setiap fungsi *residual* ditumpuk 3 lapisan konvolusi dengan ukuran 1x1, 3x3 dan 1x1. Modifikasi ini disebut *Deeper Bottleneck architecture*, dimana lapisan terakhir 1x1 yang bertanggung jawab untuk mengurangi dan meningkatkan dimensi dengan peningkatan lapisan konvolusi 3x3 *input/output* yang lebih kecil. Peningkatan kedalaman jaringan pada *deep convolutional neural network* tidak selalu menghasilkan akurasi pelatihan yang baik. Hal tersebut disebabkan oleh degradasi yang terjadi pada jaringan karena tidak semua jaringan dapat dioptimasi. Solusi permasalahan yang diusulkan oleh *ResNet* adalah dengan menerapkan *skip connection* atau

shortcut. Kelebihan model *ResNet* dibandingkan dengan model arsitektur CNN yang lain adalah kinerja dari model ini tidak menurun walaupun arsitekturnya semakin dalam, selain itu, perhitungan komputasi yang dilakukan lebih ringan dan kemampuan untuk melatih jaringan yang lebih baik. Model *ResNet* diimplementasikan dengan melakukan *skip connection* pada dua sampai tiga *layer* yang mengandung ReLU dan *batch normalization* di antara arsitekturnya. Gambar *Block ResNet* dapat dilihat pada Gambar 7.



Gambar 7. Blok *ResNet* (He *et al.*, 2016)

Blok *ResNet* dapat didefinisikan kedalam persamaan (4) berikut:

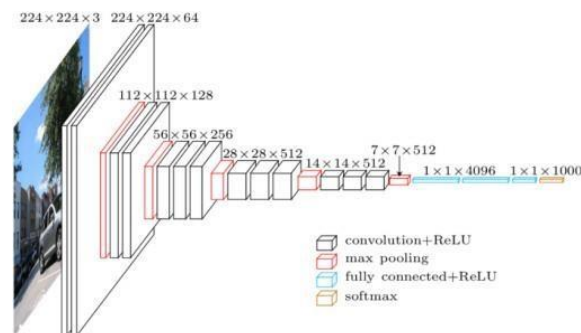
$$y = F(x, W_i + x) \dots \dots \dots (4)$$

x dan y merupakan vektor masukan dan keluaran dari *layer*, kemudian fungsi F merepresentasikan oleh *Residual* map untuk dipelajari (He *et al.*, 2016).

Residual Block pada *ResNet* dapat dilakukan apabila dimensi data masukan sama dengan dimensi data keluaran, selain itu setiap blok *ResNet* terdiri dari 2 *layer* (untuk jaringan *ResNet* 18 dan *ResNet* 34) atau 3 *layer* (untuk jaringan *ResNet* 50, 101, 152). Dua lapisan awal dari arsitektur *ResNet* menyerupai *GoogLeNet* dengan melakukan *convolution* 7×7 dan *max pooling* berukuran 3×3 dengan jumlah *stride* 2. *ResNet* memiliki beberapa macam jenis arsitektur yang dibedakan berdasarkan jumlah *layer* yang digunakan, mulai dari 18 *layer*, 34 *layer*, 50 *layer*, 101 *layer*, sampai 152 *layer* (He *et al.*, 2016).

2.6 Visual Geometry Group Network (VggNet)

VggNet atau *Visual Geometry Group Network* merupakan lapisan CNN yang disusun oleh Karen Simonyan dan Andrew Zisserman, dalam kontribusinya menunjukkan kedalaman dari jaringan untuk performa yang baik. *VggNet* memiliki beberapa jenis, dimana masing-masing jenis memiliki perbedaan pada jumlah lapisan konvolusi yang digunakan. *VggNet* terdiri dari *Vgg16* dan *Vgg19*, dimana *Vgg16* adalah model arsitektur yang terdiri dari 16 *layers* terlatih, yaitu *layer* yang mempunyai bobot (Simonyan & Zisserman, 2014). Gambar model arsitektur *Vgg16* dapat dilihat pada Gambar 8.



Gambar 8. Arsitektur *Vgg16* (Simonyan & Zisserman, 2014)

Vgg16 selama pelatihannya, menggunakan gambar 224×224 RGB ukuran tetap yang dimasukkan ke *convnet*, sebelum pemrosesan dilakukan nilai RGB yang dihitung pada setiap piksel akan dikurangi, kemudian gambar dilewatkan melalui tumpukan lapisan konvolusi, di mana menggunakan filter dengan bidang reseptif yang sangat kecil: 3×3 (yang merupakan ukuran terkecil untuk menangkap gagasan kiri/kanan, atas/bawah, tengah dan memiliki bidang reseptif efektif yang sama dengan satu 7×7), karena memiliki lebih banyak *non-linear*, dan memiliki lebih sedikit parameter salah satu konfigurasi, filter konvolusi 1×1 , yang juga digunakan dapat dilihat sebagai perubahan *linear* dari saluran input (diikuti oleh *non-linear*). Arsitektur *VggNet* lainnya dengan jumlah *layer* terbanyak saat ini adalah *Vgg19* yang secara spesifik memuat 19 lapisan utama dengan filter atau *kernel* berukuran 3×3 . 19 lapisan utama tersebut terdiri dari 16 lapisan konvolusi yang masing-masing lapisannya diikuti oleh lapisan ReLU dan 3

lapisan FC, selain itu, terdapat juga lapisan lain seperti 5 lapisan *MaxPool* dan 1 lapisan *SoftMax*. Tabel 1. yang menunjukkan lapisan *Vgg19*.

Tabel. 1. Lapisan *Vgg19* (Simonyan & Zisserman, 2014)

No.	Lapisan <i>Vgg19</i>
1.	<i>Input: Fixed Size (224, 224, 3)</i>
2.	Conv1_1 (3, 64, <i>stride</i> = 1, <i>padding</i> = 1) + ReLu
3.	Conv1_2 (64, 64, <i>stride</i> = 1, <i>padding</i> = 1) + ReLu
4.	<i>MaxPool (kernel = 2x2, stride = 2, padding = 0)</i>
5.	Conv2_1 (64, 128, <i>stride</i> = 1, <i>padding</i> = 1) + ReLu
6.	Conv2_2 (128, 128, <i>stride</i> = 1, <i>padding</i> = 1) + ReLu
7.	<i>MaxPool (kernel = 2x2, stride = 2, padding = 0)</i>
8.	Conv3_1 (128, 256, <i>stride</i> = 1, <i>padding</i> = 1) + ReLu
9.	Conv3_2 (256, 256, <i>stride</i> = 1, <i>padding</i> = 1) + ReLu
10.	Conv3_3 (256, 256, <i>stride</i> = 1, <i>padding</i> = 1) + ReLu
11.	Conv3_4 (256, 256, <i>stride</i> = 1, <i>padding</i> = 1) + ReLu
12.	<i>MaxPool (kernel = 2x2, stride = 2, padding = 0)</i>
13.	Conv4_1 (256, 512, <i>stride</i> = 1, <i>padding</i> = 1) + ReLu
14.	Conv4_2 (512, 512, <i>stride</i> = 1, <i>padding</i> = 1) + ReLu
15.	Conv4_3 (512, 512, <i>stride</i> = 1, <i>padding</i> = 1) + ReLu
16.	Conv4_4 (512, 512, <i>stride</i> = 1, <i>padding</i> = 1) + ReLu
17.	<i>MaxPool (kernel = 2x2, stride = 2, padding = 0)</i>
18.	Conv5_1 (512, 512, <i>stride</i> = 1, <i>padding</i> = 1) + ReLu
19.	Conv5_2 (512, 512, <i>stride</i> = 1, <i>padding</i> = 1) + ReLu
20.	Conv5_3 (512, 512, <i>stride</i> = 1, <i>padding</i> = 1) + ReLu
21.	Conv5_4 (512, 512, <i>stride</i> = 1, <i>padding</i> = 1) + ReLu
22.	<i>MaxPool (kernel = 2x2, stride = 2, padding = 0)</i>
23.	<i>Fully Connected (4096)</i>
24.	<i>Fully Connected (4096)</i>
25.	<i>Fully Connected (1000)</i>
26.	<i>SoftMax</i>

Tabel 1 diatas dapat dilihat bahwa *Vgg19* memiliki *input* berukuran tetap, yaitu 224x224 dengan format RGB, ketika *input* memasuki lapisan Conv1_1, ukuran *input* berubah menjadi 224x224x64. Perubahan ukuran *input* terjadi karena lapisan konvolusi Conv1_1 memiliki filter berjumlah 64 yang masing-

masing berukuran filter 3x3 dengan *stride* berukuran 1 kemudian, *input* akan masuk ke lapisan ReLu yang memiliki fungsi untuk mengganti nilai *non-linearity* milik *input* dari hasil perhitungan pada lapisan Conv1_1. Lapisan selanjutnya yang akan dilewati adalah Conv 2_2, pada lapisan ini tidak terjadi perubahan ukuran pada *input*, karena lapisan Conv 2_2 memiliki parameter masukan sebanyak 64 *channel* dan keluaran sebanyak 128 *channel*. Lapisan selanjutnya yang akan dilewati *input* adalah lapisan *MaxPool*, karena lapisan ini memiliki format *padding*, *kernel*, dan *stride* yang berbeda dengan lapisan konvolusi, maka nilai *height* dan *width input* akan berubah dari nilai awal 224 menjadi setengahnya. Proses ini akan terus berulang hingga *input* berada pada lapisan *MaxPool* sebelum lapisan *FC*, ketika *input* memasuki lapisan *FC* tersebut, maka vektor *input* akan dihubungkan satu sama lain sehingga dapat diproses pada lapisan *softmax* dan mendapatkan hasil akhir klasifikasi.

2.7 Perhitungan Akurasi

Perhitungan tingkat akurasi merupakan perbandingan antara data yang terklasifikasi benar dengan keseluruhan data. Perhitungan tingkat akurasi dihitung menggunakan rumus persamaan (5) dibawah ini (Bekkar m *et al.*, 2013).

$$Detection\ rate = \frac{Tp}{Tp+TN} \times 100 \dots\dots\dots(5)$$

Keterangan :

TP (*True Positive*) = jumlah data yang diprediksi benar

TN (*True Negatif*) = jumlah data yang diprediksi salah

2.8 Confusion Matrix

Confusion Matrix merupakan matriks yang digunakan untuk mengevaluasi model klasifikasi dengan memprediksi objek yang benar atau salah. *Confusion Matrix* direpresentasikan dengan tabel. Tabel 2. menjelaskan klasifikasi jumlah data uji yang benar dan klasifikasi jumlah data uji yang salah (Bekkar M *et al.*, 2013).

Tabel. 2. *Confusion Matrix* (Bekkar m et al., 2013).

<i>Correct Classification</i>	<i>Classified as</i>	
	<i>Predicted “+”</i>	<i>Predicted “-”</i>
<i>Actual “+”</i>	<i>True Positives</i>	<i>Negatives Actual</i>
<i>Actual “-”</i>	<i>False Positives</i>	<i>True Negatives</i>

Keterangan:

True Positives (TP) = Data positif yang terdeteksi benar

False Positives (FP) = Data negatif yang terdeteksi sebagai data positif

True Negatives (TN) = Data negatif yang terdeteksi benar

False Negatives (FN) = Data positif yang terdeteksi sebagai data negatif

Tabel *confusion matrix* bertujuan untuk menghitung kinerja metode klasifikasi dengan menghitung jumlah *recall*, *precision*, *accuracy*, dan *error rate* (Bekkar M et al., 2013).

1. *Recall*

Recall merupakan metode pengujian yang membandingkan jumlah informasi relevan yang didapatkan sistem dengan jumlah seluruh informasi relevan yang ada dalam koleksi informasi (baik yang terambil maupun yang tidak terambil oleh sistem). Persamaan *recall* dapat dilihat pada persamaan (6) berikut.

$$\text{recall} = \frac{\text{TP}}{\text{TP}+\text{FN}} \times 100\% \dots\dots\dots(6)$$

2. *Precision*

Precision adalah metode pengujian yang membandingkan jumlah informasi relevan yang didapatkan sistem dengan jumlah seluruh informasi yang terambil oleh sistem baik yang relevan maupun tidak. Persamaan *precision* dapat dilihat pada persamaan (7) berikut.

$$\text{precision} = \frac{\text{TP}}{\text{TP}+\text{FP}} \times 100\% \dots\dots\dots(7)$$

3. *Accuracy*

Accuracy adalah metode pengujian berdasarkan tingkat kedekatan antara nilai prediksi dengan nilai aktual, dengan mengetahui jumlah data yang diklasifikasikan secara benar maka dapat diketahui akurasi hasil prediksi. Persamaan *accuracy* dapat dilihat pada persamaan (8) berikut

$$\text{accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \times 100\% \dots\dots\dots(8)$$

4. *F1 Score*

F1 score adalah nilai rata-rata harmonik dari *recall* dan *precision*. *F1 score* dinyatakan dengan Persamaan (9) berikut.

$$\text{precision} = \frac{TP}{TP + \frac{1}{2}(FP+FN)} \times 100 \dots\dots\dots(9)$$

III. METODOLOGI PENELITIAN

3.1 Tempat dan Waktu Penelitian

Penelitian ini dilakukan pada tempat dan waktu yang dijelaskan sebagai berikut.

3.3.1 Tempat Penelitian

Penelitian akan dilaksanakan di Taman Kupu – Kupu Gita Persada yang berada di Jalan Wan Abdurrahman, Hutan, Kecamatan Hutan, Lampung, Indonesia dan Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung Jalan Prof. Dr Soemantri Brojonegoro No. 1, Bandar Lampung.

3.3.2 Waktu Penelitian

Waktu penelitian ini digambarkan dalam bentuk *gantt chart*. Waktu penelitian dapat dilihat pada Tabel 3.

Tabel. 3. Waktu Penelitian

Kegiatan	2023				
	Februari	Maret	April	Mei	Juni
Penulisan Bab 1-3	■	■	■		
Pengambilan data			■	■	
Pengujian Model				■	■
Analisa Hasil					■
Penulisan Bab 4-5					■

3.2 Alat dan Bahan

Alat dan bahan pendukung yang digunakan dalam penelitian ini adalah sebagai berikut:

3.3.1 Perangkat Keras (*Hardware*)

- a. Laptop dengan spesifikasi *RAM* 8.00 GB, *SSD* 256 GB, dan *Processor Intel i3-1115G4* 3.00GHz sebagai alat pendukung penelitian untuk proses pengolahan citra kupu- kupu.

3.3.2 Perangkat Lunak (*Software*)

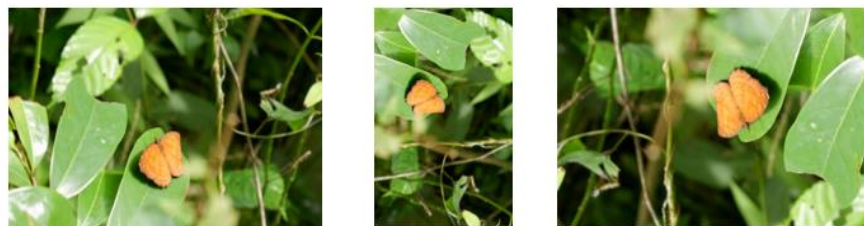
- a. Sistem Operasi *Windows 11 Home Single Language* digunakan sebagai sistem operasi pada laptop.
- b. *Google Colaboratory* dan *Google Drive* digunakan sebagai *tool* untuk melakukan proses *preprocessing* dan klasifikasi pada citra kupu-kupu.
- c. *Visual Studio code* versi 1.7.8 sebagai *text editor*
- d. *Python* versi 3.11.0 sebagai Bahasa pemrograman yang digunakan
- e. *Jupyter Notebook* versi 6.5.2 sebagai aplikasi penulisan dan eksekusi kode
- f. *Tensorflow* versi 2.12.0, *opencv* versi 4.7.0, *numpy* versi 1.22.4, dan *matplotlib* versi 3.7.1 sebagai *library* utama yang digunakan.

3.3.3 Bahan Penelitian

Bahan yang digunakan dalam penelitian ini berjumlah 800 citra dari 8 kelas kupu-kupu, terdiri dari :

- a. Kupu Jarak (*Ariadne Ariadne*),

Citra kupu jarak yang digunakan sebanyak 100 citra, Kupu jarak dapat dilihat pada Gambar 9.



Gambar 9. Kupu-kupu Jarak

b. Kupu Batik Cap (*Cethosia penthesilea*)

Citra kupu Batik Cap yang digunakan sebanyak 100 citra, Kupu Batik Cap dapat dilihat pada Gambar 10.



Gambar 10. Kupu-kupu Batik Cap

c. Kupu Hijau Biru (*Papilio peranthus*)

Citra kupu Hijau Biru yang digunakan sebanyak 100 citra, Kupu Hijau Biru dapat dilihat pada Gambar 11.



Gambar 11. Kupu-kupu Hijau Biru

d. Kupu Pantat Merah (*Pachliopta aristolochiae*)

Citra kupu Pantat Merah yang digunakan sebanyak 100 citra, Kupu Pantat Merah dapat dilihat pada Gambar 12.



Gambar 12. Kupu-Kupu Pantat Merah

e. Kupu Raja Limau (*Papilio memnon*)

Citra kupu Raja Limau yang digunakan sebanyak 100 citra, Kupu Raja Limau dapat dilihat pada Gambar 13.



Gambar 13. Kupu-kupu Raja Limau

f. Kupu Jojo (*Papilio nephelus*)

Citra kupu Jojo yang digunakan sebanyak 100 citra, Kupu Jojo dapat dilihat pada Gambar 14.



Gambar 14. Kupu-kupu Jojo

g. Kupu Harimau Kuning Hijau (*Parantica aspasia*)

Citra kupu Harimau Kuning Hijau yang digunakan sebanyak 100 citra, Kupu Harimau Kuning Hijau dapat dilihat pada Gambar 15.



Gambar 15. Kupu-kupu Harimau Kuning Hijau

h. Kupu Raja Helena (*Troides helena*).

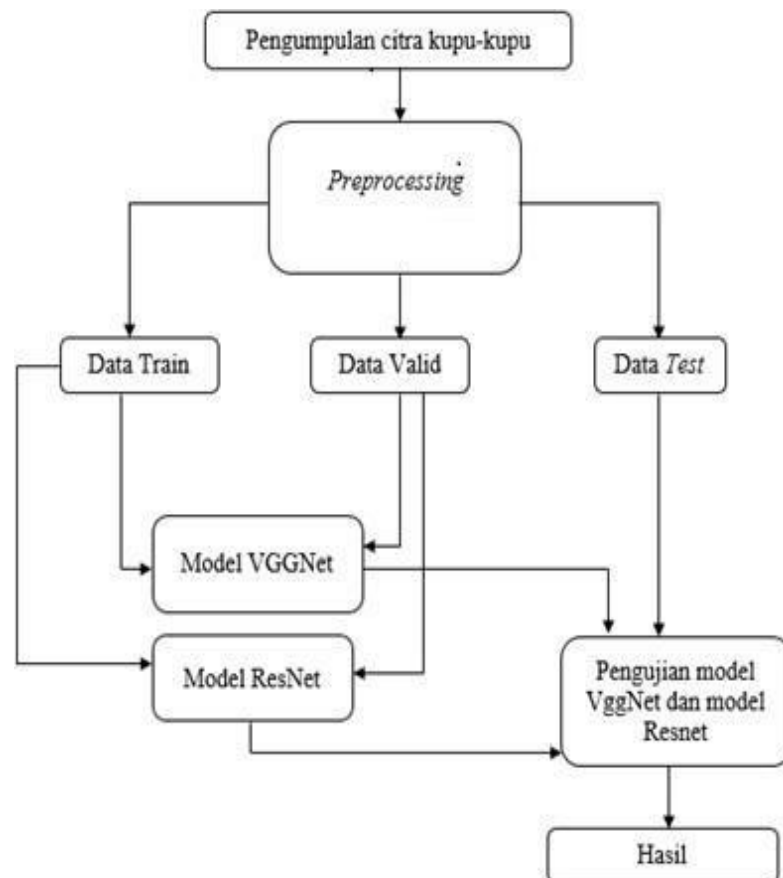
Citra kupu Harimau Raja Helena yang digunakan sebanyak 100 citra, Kupu Harimau Raja Helena dapat dilihat pada Gambar 16.



Gambar 16. Kupu-kupu Raja Helena

3.3 Tahap Penelitian

Tahap penelitian yang digunakan dalam perbandingan arsitektur *deep learning* *VggNet* dan *ResNet* dengan menginput data pada identifikasi kupu-kupu dapat dilihat pada Gambar 17.



Gambar 17. Tahap Penelitian

Tahapan yang dilakukan pada penelitian ini adalah sebagai berikut.

3.3.1 *Dataset*

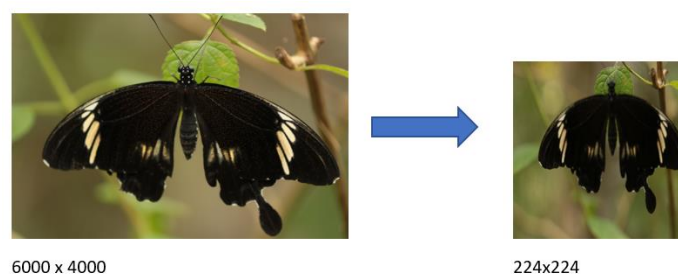
Dataset yang digunakan dalam penelitian ini diperoleh dari penangkaran kupu-kupu Taman Kupu-Kupu Gita Persada yang berlokasi di Jalan Wan Abdul Rachman, Hutan, Kecamatan Hutan, Lampung yang menjadi tempat dalam pengambilan citra kupu-kupu. Citra kupu-kupu berjumlah 800 citra untuk masing-masing kelasnya diambil menggunakan kamera Fujifilm XA-20 dengan format JPG. Jumlah dataset dapat dilihat pada Tabel 4.

Tabel. 4. Jumlah *Dataset*

Kelas	Citra
Kupu Jarak	100
Kupu Batik Cap	100
Kupu Harimau Hijau Kuning	100
Kupu Hijau Biru	100
Kupu Jojo	100
Kupu Pantat Merah	100
Kupu Raja Helena	100
Kupu Raja Limau	100
Total	800

3.3.1 *Preprocessing*

Citra masukan yang akan dilatih dan validasi perlu dilakukan *preprocessing* terlebih dahulu. *Preprocessing* yang akan dilakukan adalah dengan cara *scaling* atau merubah resolusi skala, citra yang sebelumnya berukuran besar kemudian diubah ukurannya (*resize*) menjadi 224 x 224 *pixel* dapat dilihat pada Gambar 18.



Gambar 18. Hasil *scaling*.

3.3.2 Pemisahan Data

Pemisahan data dilakukan menggunakan *train_test_split* yang terdapat pada *library scikit learn*.

a. Data latih

Data latih atau data *training* adalah data yang digunakan untuk melakukan proses *training model*. Data yang digunakan dalam penelitian ini untuk proses *training*, yaitu sebesar 70% dengan total data 560 citra kupu-kupu.

b. Data validasi

Data validasi atau data *validation* adalah data yang digunakan untuk melakukan proses validasi model dan mencegah *overfitting* dan. Data yang digunakan dalam penelitian ini untuk proses *validation*, yaitu 10% dengan total data 80 citra kupu-kupu.

c. Data uji

Data uji atau data *testing* adalah data yang digunakan untuk melakukan proses pengujian pada model. Data yang digunakan dalam penelitian ini untuk proses *testing*, yaitu sebesar 20% dengan total data 160 citra kupu-kupu.

3.3.3 Tahap Pelatihan

Tahap pelatihan model dalam penelitian ini menggunakan dua model yaitu *Vgg16* dan *ResNet50* dengan data yang akan dilatih sebanyak 800 citra kupu- kupu. Pelatihan data terhadap kedua arsitektur ini menggunakan model parameter yang sama, sehingga perbandingan bisa terlihat. Pelatihan ini menggunakan *hyperparameter* yang dapat dilihat pada Tabel 5.

Tabel. 5. Model Parameter

Nama Parameter	Nilai
<i>Input sizes</i>	224×224×3
<i>Epoch</i>	10
<i>Batch size</i>	8
<i>Number of classes</i>	8
<i>Optimizer</i>	Adam
<i>Learning rate</i>	0,001

3.3.4 Pengujian Model

Tahap ini akan dilakukan pengujian terhadap data uji yang telah diperoleh dan dilakukan proses evaluasi menggunakan parameter yang telah ditentukan, serta dilakukan perbandingan dari masing-masing arsitektur yang dipakai. Hasil dari klasifikasi akan dicatat dan dilakukan analisis pada tahap selanjutnya.

3.3.5 Hasil Pengujian

Tahap ini dilakukan pencatatan hasil dari pengujian yang didapat untuk menghitung tingkat keberhasilan model. Hasil pengujian berasal dari data latih dan data uji yang menghasilkan nilai *confusion matrix* dari nilai *recall*, *precision*, *accuracy* dan *f1 score*.

V. SIMPULAN DAN SARAN

5.1 Simpulan

Simpulan yang di peroleh dari hasil penelitian yang sudah di lakukan adalah sebagai berikut:

1. Identifikasi kupu-kupu menggunakan *Resnet* dan *VggNet* berhasil di lakukan.
2. Arsitektur *ResNet* dengan model *ResNet50* berhasil mencapai akurasi tertinggi sebesar 99,37% dengan menggunakan GPU T4 pada *Google Colaboratory* dan *jupyter notebook*.
3. Uji coba sukses dilakukan pada dua perangkat, yakni menggunakan *Google Colaboratory (online)* dan *Jupyter Notebook (offline)*.
4. Keakuratan model dalam memperkirakan kelas-kelas kupu-kupu (*precision*) dan nilai efektivitas dalam mencari data uji untuk menemukan kelas-kelas (*recall*) tertinggi oleh arsitektur *ResNet50* dengan menggunakan *Google Colaboratory* mendapatkan nilai rata-rata recall sebesar 100% dan nilai rata-rata precision sebesar 100%.
5. Jumlah parameter pada arsitektur *Vgg16* lebih besar daripada *ResNet50*, namun akurasi tertinggi dicapai oleh *ResNet50* menggunakan *Google Colaboratory* dan GPU T4, menunjukkan efektivitas kinerja *ResNet50* parameter yang lebih sedikit.

5.2 Saran

Saran yang diberikan untuk penelitian selanjutnya adalah sebagai berikut:

1. Menambah jumlah kelas dengan jenis kupu-kupu lainnya.
2. Mengembangkan klasifikasi kupu-kupu dengan memanfaatkan *hyperparameter* yang lebih beraneka ragam dari penelitian ini.
3. Mengembangkan klasifikasi kupu-kupu menggunakan model arsitektur yang lainnya seperti *ResNeXt-50*, *Inception-ResNet-V2*, *YOLO*, *Vgg19* dan lain-lain.
4. Studi ini bisa ditingkatkan dengan dibuatkan aplikasi yang berbasis *mobile* atau *web*.

DAFTAR PUSTAKA

- Achmad, A. 2017. *Potensi dan Penyebaran Kupu-Kupu di Bantimurung*. Makassar: Gramedia Pustaka utama.
- Almryad, A. S., & Kutucu, H. 2019. Automatic Identification For Field Butterflies by Convolution Neural Network. *Engineering Science and Technology an International Journal*.
- Amir, M., WA, N., & S, K. 2008. *Serangga Taman Nasional Gunung Halimun Jawa Bagian Barat*. Bogor: BCP-JICA.
- Bekkar, M., Djemaa, H. K., & Alitouche, T. A. 2013. Evaluation Measures For Models Assessment Over Imbalanced Data Sets. *Journal of Information Engineering and Application*.
- Borugadda, P., Lakshmi, R., & Govindu, S. 2021. Classification of Cotton Leaf. *Current Journal of Applied Science and Technology*.
- He, K., Zhang, X., Ren, S., & Sun, J. 2016. *Deep Residual Learning for Image Recognition*. Las Vegas: IEEE.
- Jain, R. 2018. Convolution Neural Network Alzheimer's disease classification from magnetic resonance brain image. *ScienceDirect*.
- Wang, J., & dkk. 2016. *Efficient convolution architectures for convolutional neural network*: IEEE.
- Jumuar. 2002. *Entomologi Pertanian*. Jakarta: PT. Rineka Cipta.
- Koneri, R., & Saroyo, S. 2012. Distribusi dan keanekaragaman kupu-kupu (Lepidoptera) di Gunung Manado Tua, Kawasan Taman Nasional laut Bunaken, Sulawesi Utara.
- Pranata, Y. D., & dkk. 2019. Deep Learning and SURF for Automated Classification and Detection of Calcaneus Fractures in CT Image. *Elsevier*.
- Primartha, R. 2020. *Algoritma Machine Learning*. Palembang: Gramedia Pustaka Utama.
- Rahman, M. J.-U., & dkk. 2018. *Automatic System for Detecting Invasive Ductal Carcinoma Using Convolutional Neural Network*. (p. 4). Jeju, Korea Selatan: IEEE.

- Sena, S. 2017, November 13. Pengenalan Deep Learning : Convolutional Neural Network. Indonesia.
- Simonyan, K., & Zisserman, A. 2014. *Very Deep Convolutional Networks*. Oxford University: arXiv preprint.
- Soekardi, H. A. 2016. *Kupu - Kupu Lampung Taman Kupu - Kupu Gita Persada Bandar Lampung*. Bandar Lampung, Lampung, Indonesia: Yayasan Sahabat Alam.
- Suartika, I. W., Wijaya, A. Y., & Soelaiman, R. 2016. Klasifikasi Citra Menggunakan Convolutional Neural Network pada Caltech 101. *Jurnal Teknik ITS*, Vol.5 No.1.
- Theivaprakasham, H. 2020. Identification of India Butterflies Using Deep Convolutional Neural Network. *Elsevier*.