

**PERBANDINGAN KINERJA *DEEP LEARNING LENET* DAN
MOBILENET DENGAN *AUGMENTASI DATA* PADA
IDENTIFIKASI KUPU-KUPU**

(Skripsi)

Oleh

**PANDI BAREP ARIANZA
NPM 1817051002**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2022**

**PERBANDINGAN *KINERJA DEEP LEARNING LENET* DAN
MOBILENET DENGAN *AUGMENTASI DATA* PADA**

IDENTIFIKASI KUPU-KUPU

Oleh

PANDI BAREP ARIANZA

Skripsi

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

Pada

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



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2022**

ABSTRACT

PERFORMANCE COMPARISON OF LENET AND MOBILENET DEEP LEARNING ARCHITECTURE WITH DATA AUGMENTATION ON BUTTERFLY IDENTIFICATION

By

PANDI BAREP ARIANZA

Indonesia is one of the countries with the highest level of biodiversity in the world, one of the biodiversity with the most butterfly ownership in the world. Butterflies are part of natural life that is important in maintaining ecosystems. Butterflies are active during the day and have attractive colors on their wings and are different from other butterflies. The diversity of these butterflies is quite difficult to recognize if you only look at the colors, patterns and patterns found on the wings of butterflies in distinguishing species. Deep learning is an artificial wing network model that has been widely distributed and developed in image recognition. Deep learning can be used as a technology in overcoming the problem of butterfly identification. This study aims to compare the LeNet and MobileNet architectures in the identification of butterfly objects based on the level of accuracy of the test of the two architectures. The dataset used is 800 images, then augmentation is carried out on the dataset so that the data becomes 1600 images. The tools used are Google Colab and Jupyter Notebook on the Nvidia Tesla K20. The results show that MobileNet using rotate augmentation is superior in terms of accuracy, reaching 99.37% and LeNet using rotate augmentation reaching 98.44% accuracy.

Keywords: Butterflies; Deep Learning; LeNet; MobileNet.

ABSTRAK

PERBANDINGAN KINERJA ARSITEKTUR DEEP LEARNING LENET DAN MOBILENET DENGAN AUGMENTASI DATA PADA IDENTIFIKASI KUPU-KUPU

Oleh

PANDI BAREP ARIANZA

Indonesia merupakan salah satu negara dengan tingkat keanekaragaman hayati tertinggi di dunia, salah satu keanekaragaman hayati dengan kepemilikan kupu-kupu terbanyak di dunia. Kupu-kupu merupakan bagian dari kehidupan alam yang penting dalam mempertahankan ekosistem. Kupu-kupu aktif pada siang hari dan memiliki warna yang menarik pada sayapnya dan berbeda-beda dengan kupu-kupu lainnya. Keanekaragaman kupu-kupu ini cukup sulit untuk dikenali jika hanya melihat dari warna, pola dan corak yang terdapat pada sayap kupu-kupu dalam membedakan spesies. Deep learning merupakan model jaringan sayap tiruan yang telah tersebar luas dan dikembangkan dalam pengenalan citra. Deep learning dapat digunakan sebagai teknologi dalam mengatasi permasalahan identifikasi kupu-kupu. Penelitian ini bertujuan membandingkan arsitektur LeNet dan MobileNet dalam identifikasi objek kupu-kupu berdasarkan tingkat akurasi uji kedua arsitektur. Dataset yang digunakan 800 citra, kemudian dilakukan augmentasi pada dataset sehingga data menjadi 1600 citra. Tools yang digunakan adalah Google Colab dan Jupyter Notebook pada Nvidia Tesla K20. Hasil Penelitian menunjukkan MobileNet menggunakan augmentasi rotate unggul dalam hal akurasi yaitu mencapai 99,37% dan LeNet menggunakan augmentasi rotate mencapai tingkat akurasi 98,44%.

Kata Kunci: Kupu-kupu; Deep Learning; LeNet; MobileNet.

Judul Skripsi : **PERBANDINGAN KINERJA *DEEP LEARNING LENET* DAN *MOBILENET* DENGAN *AUGMENTASI DATA* PADA IDENTIFIKASI KUPU-KUPU**

Nama Mahasiswa : **Pandi Barep Arianza**

Nomor Pokok Mahasiswa : 1817051002

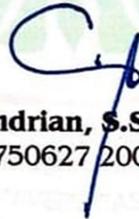
Program Studi : S1 Ilmu Komputer

Jurusan : 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 Penguji : **Rico Andrian, S.Si., M.Kom.**



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



Penguji
Bukan Pembimbing : **Didik Kurniawan, S.Si., M.T.**



2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



Dr. Eng. Supto Dwi Yuwono, S.Si., M.T.
NIP 19740705 200003 1 001



Tanggal Lulus Ujian Skripsi : **12 Agustus 2022**

PERNYATAAN

Saya bertanda tangan di bawah ini, menyatakan bahwa skripsi saya yang berjudul **“Perbandingan Kinerja Deep Learning LeNet dan MobileNet dengan Augmentasi Data pada Identifikasi Kupu-Kupu”** merupakan karya saya sendiri dan bukan karya orang lain. Semua tulisan yang tertuang di 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 saya bersedia menerima sanksi berupa pencabutan gelar yang telah saya terima.

Bandar Lampung, 17 Agustus 2022



Pandi Barep Arianza

NPM. 1817051002

RIWAYAT HIDUP



Penulis dilahirkan pada tanggal 12 Oktober 2000 di Hanura, Kecamatan Teluk Pandan, Kabupaten Pesawaran, Provinsi Lampung sebagai anak pertama dari dua bersaudara. Ayah bernama Agus Sugianto dan Ibu Sri Hartati. Penulis menyelesaikan menyelesaikan Pendidikan Sekolah Dasar (SD) di SDN 1 Hanura, Teluk Pandan, Pesawaran pada tahun

2012, menyelesaikan Sekolah Menengah Pertama (SMP) di SMPN 1 Padang Cermin pada tahun 2015, kemudian menyelesaikan sekolah di jenjang Sekolah Menengah Atas (SMA) SMAN 1 Padang Cermin dan lulus pada tahun 2018.

Pada tahun 2018, penulis terdaftar sebagai mahasiswa Jurusan Ilmu Komputer FMIPA Universitas Lampung melalui jalur SNMPTN. Selama menjadi mahasiswa, penulis melakukan beberapa kegiatan antara lain.

1. Menjadi anggota Adapter Himpunan Mahasiswa Jurusan Ilmu Komputer pada periode 2018/2019.
2. Menjadi anggota bidang Kaderisasi Himpunan Mahasiswa Jurusan Ilmu Komputer periode 2018/2019.
3. Menjadi Kepala Bidang Media Informasi Himpunan Mahasiswa Jurusan Ilmu Komputer periode 2019/2020.

4. Menjadi anggota bidang Media Informasi Badan Eksekutif Mahasiswa FMIPA UNILA priode 2020/2021.
5. Menjadi Asisten Dosen Jurusan Ilmu Komputer tahun 2019 hingga 2021.
6. Melaksanakan Kerja Praktek pada bulan Februari periode 2020/2021 di PT Arsi Enarcon.
7. Melaksanakan KKN di Desa Tanjung Agung, Kecamatan Way Lima, Kabupaten Pesawaran, Lampung pada tahun 2020/2021 dengan program kerja pelatihan desain.

MOTTO

“Jika kamu tidak sanggup menahan lelahnya belajar maka kamu harus sanggup menahan perihnya kebodohan.”

(Imam Syafi'i)

“Ilmu jangan hanya objek hafalan, ilmu untuk memahami dan menuntaskan suatu persoalan”

(Najwa shihab)

“Jika kamu tidak menyukai sesuatu, ubahlah. Jika kamu tidak bisa mengubahnya, ubahlah cara kamu memandangnya”

(Mary Engelbreit)

“Masalah-masalah bukanlah tanda untuk berhenti. Tetapi masalah itu adalah petunjuk untuk berhasil”

(author)

PERSEMBAHAN

Puji Syukur kepada Allah SWT sehingga skripsi ini dapat diselesaikan.

Skripsi yang disusun ini adalah suatu kebanggaan yang saya persembahkan kepada:

Kedua Orangtuaku yang Tercinta

Bapak dan ibu yang senantiasa memberikan segala perhatian, kasih sayang, cinta, dukungan moril maupun materil serta doa terbaik bagi kesuksesan anak-anaknya yang tidak bisa diukur dan dibalas dengan apapun.

Teman seperjuangan dan keluarga besar yang senantiasa memberikan dukungan, kasih sayang, semangat, dan menanyakan kapan skripsi ini akan selesai.

Keluarga Besar Ilmu Komputer 2018

Serta Almamater Tercinta,

UNIVERSITAS LAMPUNG

SANWACANA

Puji Syukur penulis panjatkan terhadap Allah SWT sehingga penulis dapat menyelesaikan skripsi di Jurusan Ilmu Komputer Fakultas MIPA Universitas Lampung dengan judul “Perbandingan Kinerja Deep Learning LeNet dan MobileNet dengan Augmentasi Data pada Identifikasi Kupu-Kupu”. Penulis mengucapkan terima kasih yang paling tulus dan sebesar-besarnya kepada semua pihak yang telah membantu dalam pelaksanaan penelitian dan penyusunan laporan skripsi ini, antara lain:

1. Kedua orang tua yang sangat penulis sayangi, Bapak dan Ibu yang senantiasa mengajarkan kebaikan, memberikan perhatian, kasih sayang, do’a terbaik, dukungan, serta kepercayaan atas setiap keputusan yang diambil oleh penulis hingga detik ini.
2. Saudara dan saudari penulis, Bisma Septian dan Alfa Rensy Inada (alm) yang selalu memberikan perhatian, pengertian, dukungan, dan semangat dalam setiap langkah penyusunan skripsi ini hingga saat ini.
3. Bapak Rico Andrian, S.Si., M.Kom. selaku pembimbing utama penulis dalam penelitian ini yang senantiasa memberikan arahan, bantuan ketika mengalami kesulitan selama penelitian, semangat, dan motivasi terbaik dalam menyelesaikan penelitian ini.

4. Bapak Dr. Eng. Suropto Dwi Yuwono, M.T selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
5. Bapak Didik Kurniawan, S.Si., M.T selaku Ketua Jurusan dan pembimbing akademik yang telah memperhatikan serta memberikan motivasi kepada penulis dalam menempuh pendidikan di Ilmu Komputer Fakultas MIPA Universitas Lampung.
6. Bapak Dr. rer. nat. Akmal Junaidi, M.Sc selaku Sekretaris Jurusan Ilmu Komputer Fakultas MIPA Universitas Lampung.
7. Seluruh Dosen Jurusan Ilmu Komputer yang telah memberikan ilmu dan pelajaran terbaik selama penulis menempuh pendidikan di Jurusan Ilmu Komputer Universitas Lampung.
8. Seluruh Karyawan dan Staf Jurusan Ilmu Komputer Universitas Lampung yang telah memberikan bantuan dalam banyak hal selama ini.
9. Teman-teman masa kecil Muhammad Ramadani, Hasannudin, Mega Andesta, dan Salsabila yang telah memberikan arahan, dukungan dan semangat kepada penulis untuk menyelesaikan studi ini.
10. Mantan terbaik Renyda Murni Syafriaki yang selalu membantu, mendukung, dan mengingatkan penulis ketika malas mengerjakan revisian skripsi.
11. Sahabat terbaik penulis Dara Arka Fidela yang selalu mendukung, menemani, dan berbagi cerita indah selama masa kuliah.
12. Anggota Rich Arbi Hidayatullah, Abie Perdana Kusuma, Yuan Ferdinand A. Kailuhu, Rika Ningtias Azhari, Syela Septania dan Annisa Nurwalikadani yang selalu ada saat kuliah di jurusan ilmu komputer.

13. Teman pertama di jurusan ilmu komputer Arfina Shella Meilany tempat berbagi cerita disaat penulis merasa sedih.
14. Seluruh Anggota SPARTAN Atikah Istiqomah, Dian Indah Lestari, Eggi Amandara Sari, Reda Meiningtyas, Noverina Ika Tama, Vio Heven Susanto, Rochmat Ali, Kasandra Cahyani, Hamid Al Malik, Zuhri Nopriyanto, teman berbagi keluh kesah selama bimbingan.
15. Keluarga Ilmu Komputer 2018 yang tidak bisa penulis sebut satu persatu. Keluarga kedua penulis, rekan kelompok, rekan diskusi, rekan bercanda, dan telah memberi arti dan warna serta pengalaman tak ternilai semasa duduk di bangku kuliah.
16. Teman-teman Himakom dan Brighter yang sudah mengajarkan banyak hal dalam berorganisasi, memberikan banyak pengalaman.

Penulis menyadari bahwa penyusunan skripsi ini masih jauh dari kata sempurna. Namun penulis sangat mengharapkan skripsi ini dapat bermanfaat bagi para civitas akademik Universitas Lampung pada umumnya dan mahasiswa Ilmu Komputer pada khususnya.

Bandar Lampung, 17 Agustus 2022



Pandi Barep Arianza

NPM. 1817051002

DAFTAR ISI

	Halaman
DAFTAR TABEL	iii
DAFTAR GAMBAR	iv
I. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	5
1.3 Batasan Masalah.....	5
1.4 Tujuan Penelitian.....	5
1.5 Manfaat Penelitian.....	6
II. TINJAUAN PUSTAKA	7
2.1 Kupu-kupu.....	7
2.2 <i>Deep Learning</i>	8
2.3 <i>Convolutional Neural Network</i>	9
2.4 <i>LeNet</i>	12
2.5 <i>MobileNet</i>	13
2.6 <i>Preprocessing</i>	15
2.7 <i>Augmentasi Data</i>	15
2.8 <i>Confusion matrix</i>	17
2.9 <i>Hyperparameter</i>	19
2.10 <i>Loss Function</i>	20
III. METODOLOGI PENELITIAN	21
3.1 Tempat dan Waktu Penelitian	21
3.2 Alat dan Bahan	21
3.3 Tahapan Penelitian	22

IV. HASIL DAN PEMBAHASAN	27
4.1 Klasifikasi Arsitektur <i>LeNet</i>	27
4.2 Klasifikasi Arsitektur <i>MobileNet</i>	32
4.3 Pembahasan	35
V. SIMPULAN DAN SARAN	45
5.1 Simpulan.....	45
5.2 Saran.....	46
DAFTAR PUSTAKA	47

DAFTAR TABEL

Tabel	Halaman
1. <i>Confusion matrix</i> . (Han & Kamber, 2011).....	17
2. Jumlah <i>dataset</i>	24
3. Jumlah <i>dataset</i> setelah <i>augmentasi</i>	24
4. <i>Hyperparameters</i>	26
5. Hasil akurasi arsitektur <i>LeNet</i> dengan <i>google colab</i>	29
6. Hasil akurasi arsitektur <i>LeNet</i> dengan <i>tesla k20</i>	30
7. Hasil akurasi arsitektur <i>MobileNet</i> dengan <i>google colab</i>	33
8. Hasil akurasi arsitektur <i>MobileNet</i> dengan <i>tesla k20</i>	33
9. <i>Confusion matrix</i> arsitektur <i>LeNet rotate</i> dengan mesin <i>colab</i>	36
10. <i>Recall, precision, f1 score</i> arsitektur <i>LeNet rotate</i> mesin <i>colab</i>	38
11. <i>Confusion matrix</i> arsitektur <i>MobileNet rotate</i> dengan mesin <i>tesla</i>	42
12. <i>Recall, precision, f1 score</i> arsitektur <i>MobileNet rotate</i> dengan mesin <i>tesla</i>	43

DAFTAR GAMBAR

Gambar	Halaman
1. Tubuh kupu-kupu.....	8
2. Model CNN (Suryanto,2018).....	9
3. <i>Max pooling</i> dan <i>average pooling</i> (Pangestu, <i>et al.</i> , 2020).	10
4. Proses ReLU (Pangestu, <i>et al.</i> , 2020).	11
5. Arsitektur <i>LeNet</i>	12
6. <i>Depthwise separable convolutions</i> (Alwanda, <i>et al.</i> ,2020).....	14
7. <i>Augmentasi data</i>	15
8. Teknik <i>rotation</i>	16
9. Teknik <i>color space transformation</i>	16
10. Teknik <i>flipping</i>	17
11. Tahapan penelitian.	23
12. Model <i>LeNet</i>	28
13. Grafik <i>training</i> model <i>LeNet</i> kategori tanpa <i>augmentasi</i> dengan <i>google colab</i>	30
14. Grafik <i>training</i> model <i>LeNet</i> kategori <i>rotate</i> dengan <i>tesla k20</i>	31
15. Model <i>MobileNet</i>	32
16. Grafik <i>training</i> model <i>MobileNet</i> kategori tanpa <i>augmentasi</i> dengan <i>google colab</i>	34
17. Grafik <i>training</i> model <i>MobileNet</i> kategori <i>rotate</i> dengan <i>tesla k20</i>	34
18. Kemiripan kupu raja helena dan kupu pantat merah.....	39
19. Kemiripan kupu harimau hijau kuning dan kupu raja helena.	40
20. Kemiripan kupu jarak dan kupu pantat merah.	40
21. Kemiripan kupu raja helena dan kupu jarak	41
22. Kemiripan kupu jarak dan kupu raja helena.	44

I. PENDAHULUAN

1.1 Latar Belakang

Indonesia merupakan salah satu negara dengan tingkat kelimpahan keanekaragaman hayati tertinggi di dunia, salah satu keanekaragaman hayati dengan kepemilikan kupu-kupu terbanyak di dunia, terdapat sebanyak 2.500 jenis kupu-kupu yang ada di Indonesia dan 50% dari total jumlah kupu-kupu tersebut endemik yang berarti hanya ada di tempat itu saja. Kupu-kupu merupakan bagian dari kehidupan di alam yang sangat penting dalam mempertahankan ekosistem, yaitu sebagai salah satu serangga penyerbuk pada proses pembuahan pada tumbuhan berbunga. Serangga ini memiliki dua pasang sayap dan sayap belakang biasanya sedikit lebih kecil dari pada sayap depan. Kupu-kupu biasanya aktif pada siang hari dan memiliki sayap yang indah dengan warna yang menarik dan berbeda-beda dari kupu-kupu lainnya, hal ini juga yang menjadikan kupu-kupu sebagai serangga unik. Serangga unik ini memiliki keragaman dari jenis, corak, warna, dan pola.

Keanekaragaman kupu-kupu berbeda-beda di setiap tempat, hal ini terjadi karena keberadaan kupu-kupu di suatu habitat sangat erat dengan faktor lingkungan dan dijadikan sebagai bioindikator perubahan kualitas lingkungan, karena kupu-kupu sangat sensitif terhadap perubahan ekosistem (Soekardi, *et al.*, 2016). Pulau Sumatra merupakan salah satu daerah dengan faktor lingkungan yang berbeda-beda, perbedaan faktor inilah yang menyebabkan jenis kupu-kupu di setiap daerah juga berbeda-

beda (Koneri & Saroyo, 2012). Pulau Sumatera diperkirakan terdapat sekitar 1.000 spesies kupu-kupu dan di Taman Nasional Bukit Barisan Selatan terdapat 185 spesies, Taman Nasional Way Kambas terdapat sekitar 77 spesies, dan Taman kupu-kupu Gita Persada, Gunung Betung Lampung terdapat 107 spesies (Soekardi, 2007).

Taman Kupu-Kupu Gita Persada merupakan salah satu penangkaran kupu-kupu untuk melindungi kelestariannya dari ancaman kepunahan spesies. Taman Kupu-Kupu Gita Persada telah berhasil melestarikan lebih dari 180 spesies di penangkarannya. Taman Kupu-Kupu Gita Persada terdapat banyak spesies kupu-kupu yang dikembangbiakan, namun keanekaragaman kupu-kupu ini cukup sulit dikenali jika hanya melihat dari warna, pola dan coraknya dengan kasat mata. Manusia pada dasarnya dapat membedakan spesies kupu-kupu menggunakan mata, namun masih tergolong sulit karena mata manusia memiliki keterbatasan dalam membedakan tekstur dan warna yang khas pada spesies kupu-kupu (Arzar, *et al.*, 2019). Orang awam dalam membedakan spesies kupu-kupu tergolong sangat sulit, karena kurangnya pengetahuan tentang spesies kupu-kupu, oleh karena itu, dibutuhkan sebuah teknologi yang memudahkan pengguna untuk 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 tersebar luas dan dikembangkan lebih lanjut dalam pengenalan citra, karena perkembangan teknologi. *Deep learning* dan *artificial intelligence* termasuk ke dalam bidang *computer vision* dalam penggalian informasi dari gambar. *Computer vision* mengadaptasi kemampuan penglihatan manusia dalam mengambil informasi dari objek gambar (Borugadda, *et al.*, 2021). *Deep learning* menggunakan konsep

cara kerja dari jaringan saraf otak manusia dalam menangkap objek gambar (Almryad & Kutucu, 2020). *Deep Learning* dapat memiliki tingkat akurasi yang tinggi, salah satu teknik *deep learning* yang paling banyak digunakan adalah *Convolutional Neural Network* (CNN). *Convolutional Neural Network* merupakan pengembangan lebih lanjut dari metode *Multilayer Perceptron* (MLP). Metode *convolutional neural network* ini memiliki jaringan yang dalam dan dapat mencapai akurasi yang tinggi serta hasil yang baik (Eka Putra, 2016).

Convolutional Neural Network dalam prosesnya terbagi menjadi *feature extraction layer* dan *fully-connected layer* (Borugadda, et al., 2021). *Feature extraction layer* digunakan untuk mengambil ciri dari citra lalu menyimpannya untuk digunakan pada tahap selanjutnya yaitu *fully-connected layer*. Tahap *fully-connected layer* merupakan tahap dalam proses pengklasifikasian objek pada citra. *Convolutional Neural Network* memiliki beberapa macam arsitektur diantaranya *LeNet* dan *MobileNet*. *LeNet* adalah arsitektur konvolusi pertama yang terdiri dari dua lapisan konvolusi dengan *ReLU*, *Pooling layer* dan *convolutional layer* lainnya (Krishna & Kalluri, 2019). *MobileNet* adalah salah satu arsitektur yang dapat mengurangi ukuran model dan membagi lapisan menjadi dua bagian yaitu *standart convolution* dan *depthwise convolution* (Venkateswarlu, et al., 2020). *Depthwise convolution* digunakan untuk menerapkan satu filter ke setiap saluran *input* dan bertindak sebagai pemfilteran lapisan, sedangkan *pointwise convolution* digunakan untuk membuat kombinasi linier dari *output* dengan bantuan 1×1 konvolusi (Pan, et al., 2020).

Penelitian terdahulu terkait perbandingan arsitektur pada citra kupu-kupu, dengan membandingkan arsitektur *VGGNet* dan arsitektur *ResNet* untuk identifikasi kupu-kupu. *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. Penelitian ini menghasilkan tingkat akurasi pada arsitektur *VGGNet* yaitu 79,5% dan arsitektur *ResNet* yaitu 70,2% (Almryad & Kutucu, 2020).

Penelitian terdahulu terkait identifikasi pada citra kupu-kupu dengan menggunakan fitur deteksi tepi (*edge detection*) dan klasifikasi *K-Nearest Neighbor* (KNN). *Dataset* yang digunakan dalam penelitian ini berjumlah 600 citra dengan 6 kelas dan masing-masing kelas berjumlah 100 citra kupu-kupu. *Dataset* dibagi menjadi *data training* sebesar 70% dan *data testing* sebesar 30%. Penelitian ini menghasilkan tingkat akurasi terhadap citra kupu-kupu sebesar 80% (Andrian, *et al.*, 2019).

Penelitian lainnya terkait identifikasi pada citra kupu-kupu dengan menggunakan fitur ekstraksi gray level co-occurrence matrix (glcm) dan klasifikasi *K-Nearest Neighbor* (KNN). *Dataset* yang digunakan dalam penelitian ini berjumlah 600 citra dengan 6 kelas dan masing-masing kelas berjumlah 100 citra kupu-kupu. *Dataset* dibagi menjadi *data training* sebesar 70% dan *data testing* sebesar 30%. Penelitian ini menghasilkan tingkat akurasi sebesar 91,1% (Andrian, *et al.*, 2019).

Penelitian sebelumnya terkait proses mengidentifikasi jenis kupu-kupu menggunakan *Convolutional Neural Network* (CNN). Penelitian ini menerapkan arsitektur *LeNet* dan *VGG-16* dalam prosesnya dan menggunakan *optimizer* Adam, Adagrad dan SGD. *Dataset* yang digunakan sebanyak 5455 citra kupu-kupu dengan 50 spesies dan dibagi menjadi 4955 *data train*, 250 *data test* dan 250 *data valid*, kemudian *dataset* dilakukan *augmentasi* pada *data train*. Tingkat akurasi yang dihasilkan pada penelitian ini menggunakan *VGG-16* sebesar 93% dan menggunakan *LeNet* sebesar 67% (Michael & Hartati, 2022).

Penelitian yang akan dilakukan yaitu perbandingan kinerja arsitektur *deep learning LeNet* dan *MobileNet* dengan *augmentasi data* pada identifikasi kupu-kupu. 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 identifikasi kupu-kupu menggunakan arsitektur *LeNet* dan *MobileNet*.

1.3 Batasan Masalah

Batasan masalah dalam penelitian ini adalah sebagai berikut :

1. Penelitian dilakukan hanya menggunakan arsitektur *LeNet* dan *MobileNet* untuk mengidentifikasi kupu-kupu.
2. Menggunakan *dataset* Kupu-Kupu berjumlah 800 citra dengan 8 kelas, yaitu *Ariadne ariadne*, *Cethosia penthesilea*, *Papilio peranthus*, *Pacliopta aristolochiae*, *Papilio memnon*, *Papilio nephelus*, *Parantica aspiasa*, dan *Troides Helena*.
3. Menggunakan *augmentasi data* jenis *rotate* pada *dataset* kupu-kupu.
4. Menggunakan *hyperparameter* dan perlakuan yang sama untuk kedua arsitektur.

1.4 Tujuan Penelitian

Tujuan dari penelitian adalah membandingkan arsitektur *LeNet* dan *MobileNet* dalam identifikasi objek kupu-kupu berdasarkan tingkat akurasi uji kedua arsitektur .

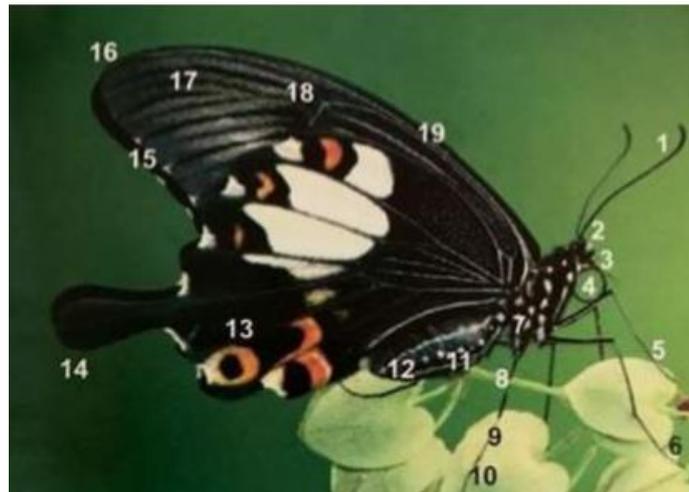
1.5 Manfaat Penelitian

Manfaat dari penelitian ini adalah dapat dijadikan sebagai bahan rujukan penelitian lainnya mengenai identifikasi kupu-kupu dengan *augmentasi data* menggunakan metode *deep learning*.

II. TINJAUAN PUSTAKA

2.1 Kupu-kupu

Kupu-kupu termasuk kedalam 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.

Bagian Kupu-kupu pada Gambar 1:

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

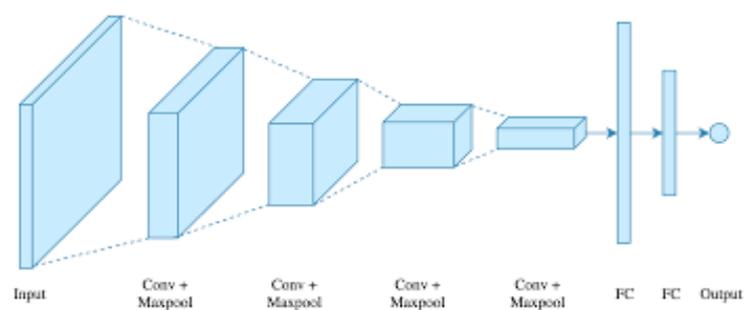
2.2 *Deep Learning*

Deep Learning merupakan cabang dari *Machine Learning* yang terinspirasi dari *kortex* pada otak manusia dengan menerapkan 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 data yang besar dan melalui banyak lapisan untuk

membaca, menyimpan, memproses, dan mengklasifikasi sebuah masukan, sehingga seringkali kinerja deep learning melampaui 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 pengembangan dari *Multi Layer Perceptron* (MLP) yang digunakan dalam mengelola data dua dimensi (Almryad & Kutucu, 2020). CNN termasuk kedalam *Deep Neural Network* yang banyak digunakan pada citra dengan jaringan yang tinggi. CNN bekerja secara otomatis dalam pengambilan ciri sebuah objek, begitu juga dalam pembentukan angka-angka *pixel* untuk menyimpan ciri dari gambar. CNN terdiri dari lapisan masukan (*input layer*), lapisan keluaran (*output layer*) dan sejumlah lapisan tersembunyi (*hidden layer*). Lapisan tersembunyi ini berisi *convolutional layers*, *pooling layers*, dan *fully connected layers* (Suyanto, 2018). Lapisan pada metode *Convolutional Neural Network* dapat dilihat pada Gambar 2.



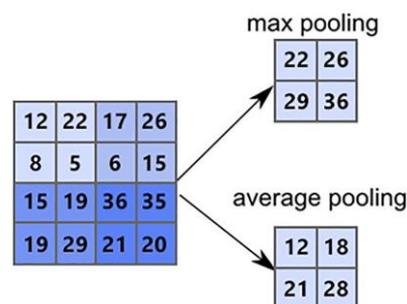
Gambar 2. Model CNN (Suryanto,2018).

2.3.1 Convolutional Layers

Convolutional layers adalah lapisan pertama yang digunakan untuk memproses citra yang masuk sehingga *layer* ini sangat penting pada metode *convolutional neural network* (Pangestu, *et al.*, 2020). Lapisan ini termasuk blok utama pada CNN yang didalamnya terdiri dari filter-filter yang dipelajari secara acak dalam melakukan operasi konvolusi yang bertujuan sebagai ekstraksi fitur untuk mempelajari representasi fitur dari *input layer*.

2.3.2 Pooling Layers

Pooling layers adalah lapisan yang berfungsi untuk mengurangi ukuran spasial dari fitur konvolusi sehingga dapat mengurangi sumber daya komputasi yang dibutuhkan untuk memproses data, melalui pengurangan dimensi dari *feature map* sehingga mempercepat komputasi, karena parameter diperbarui semakin sedikit (Pangestu, *et al.*, 2020). *Pooling layer* ada dua macam yaitu *max pooling* dan *average pooling*. Contoh *max pooling* dan *average pooling* dapat dilihat pada Gambar 3.



Gambar 3. *Max pooling* dan *average pooling* (Pangestu, *et al.*, 2020).

2.3.3 Fully Connected Layers

Fully connected layer adalah lapisan yang digunakan untuk melakukan transformasi pada dimensi data agar data dapat diklasifikasi secara linear, untuk mendapatkan hasil keluaran dari *layer* ini tidak dibutuhkan operasi konvolusi, tetapi menggunakan komputasi perkalian matriks yang diikuti dengan *bias offset*, dengan penggunaan operasi tersebut, setiap neuron memiliki koneksi penuh ke semua aktivasi dalam lapisan sebelumnya (Pangestu, *et al.*, 2020).

2.3.4 ReLU (Rectified Linear Units)

Rectified linear units (ReLU) merupakan fungsi sebagai pengubah atau menghilangkan nilai negatif pada citra. Perubahan tersebut bertujuan untuk meningkatkan kualitas citra untuk meminimalisir kesalahan (Pangestu, *et al.*, 2020). Fungsi aktivasi ReLU yaitu mengganti nilai negatif dalam citra dengan nilai 0 seperti pada Gambar 4.

1	-3	2	4	ReLU →	1	0	2	4
1	3	-2	4		1	3	0	4
5	7	-6	8		5	7	0	8
-5	7	6	-8		0	7	6	0

Gambar 4. Proses ReLU (Pangestu, *et al.*, 2020).

2.3.5 Softmax

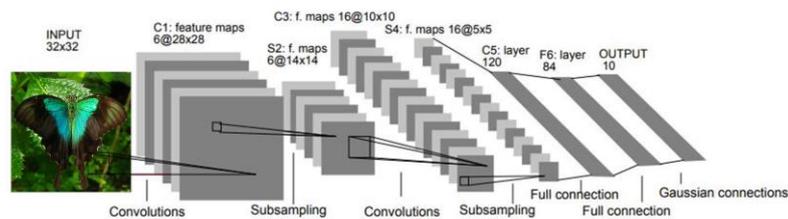
Softmax merupakan fungsi yang mengambil inputan *vector* dari bilangan real K , dan menormalisasi menjadi probabilitas terdistribusi yang terdiri dari probabilitas K , sebelum menerapkan *softmax*, beberapa komponen *vector* bisa negatif atau lebih besar dari satu dan mungkin tidak berjumlah 1, tetapi

setelah menerapkan *softmax* setiap komponen akan berada dalam interval (0-1) dan komponen akan bertambah hingga 1, sehingga dapat diartikan sebagai probabilitas, komponen *input* yang lebih besar akan sesuai dengan probabilitas yang lebih besar (Pangestu, *et al.*, 2020). Persamaan 1 pada fungsi aktivasi *softmax* sebagai berikut.

$$\rho(x) = \frac{e^x}{\sum_{k=1}^k e^x} \dots\dots\dots(1)$$

2.4 *LeNet*

LeNet adalah suatu jaringan yang memiliki lapisan banyak berbasis *Convolutional Neural Network (CNN)* pertama kali yang dikenalkan oleh Yann LeCun. *LeNet* terbentuk dari pengembangan – pengembangan versi *LeNet* sebelumnya, sehingga *LeNet* ini mempunyai jumlah lapisan yang lebih banyak daripada versi *LeNet* sebelumnya. Seiring perkembangan dan majunya kecepatan teknologi dari komputer, maka lapisan *LeNet* ini mengalami penambahan sehingga komputer dapat lebih cepat melakukan komputasi atau perhitungan matematis secara cepat (Fitriati, 2016). Arsitektur *LeNet* dapat dilihat pada Gambar 5.



Gambar 5. Arsitektur *LeNet*.

Input pada *LeNet* adalah gambar dengan ukuran 32x32x3 melalui lapisan *convolutional* dengan 6 *feature maps* dengan ukuran *filter* 5x5 dengan

satu *stride*. 6 *feature maps* ini adalah *channel* dari gambar yang sudah dilakukan operasi konvolusi dengan setiap ukuran $28 \times 28 \times 6$. *Stride* ini digunakan untuk mengontrol seberapa besar pergeseran dari suatu filter pada lapisan saat melewati *kernel size*. Lapisan kedua (S2) merupakan bagian *pooling layer* dengan filter 2×2 , 6 *feature maps* dan dua *stride*. Lapisan ini masih sama dengan lapisan sebelumnya menggunakan fungsi aktivasi *ReLU*.

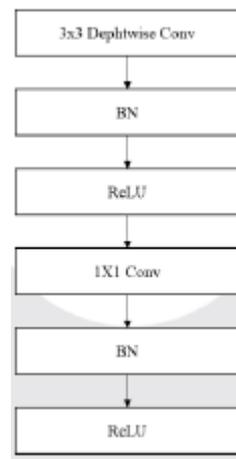
Lapisan *convolutional* kedua dengan 16 *feature maps* yang memiliki ukuran filter 5×5 dengan aktivasi *ReLU* dan satu *stride* dan didapatkan dimensi gambar dengan ukuran $10 \times 10 \times 16$. Lapisan keempat (S4) merupakan *pooling layer* dengan jenis *average pooling layer* atau *max pooling layer* yang memiliki filter 2×2 dengan aktivasi *ReLU* dan dua *stride*. Lapisan ini terdapat 400 *nodes* yang akan dihubungkan dengan dimensi gambar $5 \times 5 \times 16$.

Lapisan kelima (C5) merupakan *fully connected layer* dengan *feature maps* pada masing-masing ukuran dengan fungsi *ReLU*. 120 *nodes* pada masing-masing lapisan telah terhubung ke semua 400 *nodes* yang ada pada lapisan keempat (S4). Lapisan keenam (F6) merupakan *fully connected layer* dengan *nodes* yang berjumlah sebanyak 84 dan mendapatkan parameter latih sebanyak 10164 *nodes*. Lapisan terakhir atau *output layer* merupakan *fully connected layer* yang menggunakan aktivasi *softmax* dengan *size* berdasarkan hasil *output* gambar yang akan diklasifikasikan.

2.5 *MobileNet*

MobileNet merupakan model yang memiliki ukuran kecil baik dari jumlah parameter maupun ukuran model yang dihasilkan. Seperti namanya, *Mobile*, para peneliti dari *Google* membuat arsitektur CNN

yang dapat digunakan untuk kebutuhan *mobile*. *MobileNet* adalah sebuah arsitektur model yang dikembangkan untuk pengembangan aplikasi pada perangkat *mobile* ataupun perangkat lain yang memiliki keterbatasan sumber daya perangkat keras dengan mengurangi ukuran dan kompleksitas model menggunakan *depthwise separable convolutions* (Alwanda, *et al.*,2020). Perbedaan mendasar antara arsitektur *MobileNet* dan arsitektur CNN pada umumnya adalah penggunaan lapisan atau *layer* konvolusi dengan ketebalan *filter* yang sesuai dengan ketebalan *input image*. Gambar 6 menunjukkan gambar *depthwise separable convolutions*, yakni *blok layer* yang tersusun dari *depthwise convolution* dan *pointwise convolution*, masing-masing *layer* tersebut diikuti oleh *batch normalization* dan *ReLU nonlinearity* berikut adalah gambar arsitektur *depthwise separable convolutions*.



Gambar 6. *Depthwise separable convolutions* (Alwanda, *et al.*,2020).

Blok layer tersebut kemudian disusun secara berulang membentuk arsitektur *MobileNet*. Berdasarkan pustaka *keras.MobileNet* memiliki jumlah *layer* sebanyak 28 *convolution layer* dan 1 *fully connected layer* yang diikuti oleh lapisan *softmax* (Albawi, Mohammed, & A-Zawi, 2017).

2.6 *Preprocessing*

Preprocessing merupakan sebuah proses yang dilakukan untuk memperhalus suatu data sebelum data tersebut diolah dan berfungsi untuk meningkatkan kualitas suatu data (Sihombing, *et al.*, 2020).

2.7 *Augmentasi Data*

Augmentasi data merupakan teknik untuk memanipulasi sebuah data tanpa kehilangan makna utama atau inti dari data tersebut (W, Jondri, & Rizal, 2021). *Augmentasi data* dapat meningkatkan variasi sampel data pelatihan dan mencegah terjadinya *overfitting* (Takahashi, *et al.*, 2015). *Overfitting* adalah kondisi model bekerja sangat baik dalam mempelajari objek pada proses pelatihan, sedangkan memberikan hasil yang buruk pada proses pengujian. *Augmentasi data* dapat mencakup serangkaian teknik yang mampu meningkatkan ukuran dan kualitas data pelatihan sehingga model yang dihasilkan lebih baik dari sebelumnya (Shorten & Khosgoftaar, 2019). *Augmentasi data* pada prosesnya hanya mengembalikan data hasil *augmentasi* kedalam mesin, tetapi data asli tidak ikut dikembalikan. Teknik ini akan menghasilkan data berupa gambar yang telah diproses menggunakan parameter tertentu seperti *rotation*, *zoom*, *width_shift*, *height_shift*, *shear*, *horizontal_flip*, dan *vertical_flip*. Contoh *Augmentasi data* dapat dilihat pada Gambar 7.



Gambar 7. *Augmentasi data*.

Model yang telah diberikan data baru dengan sedikit modifikasi untuk diuji, dapat otomatis mempelajari dengan baik ketika variasi data ditambahkan pada proses pelatihan (Shorten & Khosgoftaar, 2019). *Augmentasi data* yang sering digunakan yaitu, *rotation*, *color space transformations*, *flipping* dan sebagainya.

2.7.1 *Rotation*

Rotation (rotasi) merupakan teknik dalam *augmentasi data* untuk memutar gambar. Gambar dapat diputar dengan nilai 1° - 359° baik ke arah kanan maupun kiri (Shorten & Khosgoftaar, 2019). Hasil dari teknik *rotation* dapat dilihat pada Gambar 8.



Gambar 8. Teknik *rotation*.

2.7.2 *Color space transformation*

Color space transformation atau perubahan komposisi warna digunakan untuk menambah variasi data dengan komposisi warna yang berbeda dari tiap gambar. Hasil dari teknik perubahan warna dapat dilihat pada Gambar 9.



Gambar 9. Teknik *color space transformation*.

2.7.3 Flipping

Flipping merupakan teknik dalam *augmentasi data* untuk membalik gambar secara vertikal maupun horizontal. Teknik ini termasuk mudah diterapkan dan mampu meningkatkan kinerja model klasifikasi pada saat proses pengujian (Shorten & Khosgoftaar, 2019). Hasil dari teknik *flipping* dapat dilihat pada Gambar 10.



Gambar 10. Teknik *flipping*.

2.8 Confusion matrix

Confusion matrix adalah salah satu metode untuk menghitung tingkat akurasi. *Confusion matrix* merupakan metode yang sangat berguna untuk menganalisis seberapa baik *classifier* mengenali sebuah data termasuk ke dalam kategori data uji benar dan kategori data uji salah.

Tabel 1. *Confusion matrix*. (Han & Kamber, 2011)

<i>Actual Class</i>	<i>Predicted Class</i>		Total
	<i>Yes</i>	<i>No</i>	
<i>Yes</i>	<i>TP</i>	<i>FN</i>	<i>P</i>
<i>No</i>	<i>FP</i>	<i>TN</i>	<i>N</i>
Total	<i>P'</i>	<i>N'</i>	<i>P + N</i>

Keterangan :

- a. TP (*True Positive*), jumlah data dengan nilai sebenarnya positif dan nilai prediksi positif.
- b. FP (*False Positive*), yaitu jumlah data dengan nilai sebenarnya negatif dan nilai prediksi positif.
- c. FN (*False Negative*), yaitu jumlah data dengan nilai sebenarnya positif dan nilai prediksi negatif.
- d. TN (*True Negative*), yaitu jumlah data dengan nilai sebenarnya negatif dan nilai prediksi negatif.

Nilai dari TP dan TN memberikan informasi ketika *classifier* benar dalam melakukan klasifikasi data, sedangkan nilai FP dan FN memberikan informasi ketika *classifier* salah dalam melakukan klasifikasi data, berdasarkan penelitian sebelumnya, parameter *confusion matrix* yang digunakan untuk mengevaluasi kinerja model adalah *accuracy*, *recall*, *precision*, dan *f1 score* (Yuzkat, et al., 2021).

2.8.1 Accuracy

Accuracy (akurasi) adalah matrik yang paling umum untuk melakukan evaluasi klasifikasi. Akurasi bekerja dengan cara menghitung nilai probabilitas berdasarkan nilai yang benar dari label kelas. Akurasi memiliki Persamaan 2, sebagai berikut (Bekkar et al., 2013).

$$\text{accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \times 100\% \dots\dots\dots(2)$$

2.8.2 Recall

Recall atau *sensitivity* adalah keakuratan dari nilai positif yang ada. *Recall* memiliki Persamaan 3, sebagai berikut (Bekkar et al., 2013).

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \times 100\% \dots\dots\dots(3)$$

2.8.3 Precision

Precision adalah nilai kebenaran dari prediksi yang dilakukan oleh classifier dengan label kelas yang sudah ada. *Precision* memiliki Persamaan 4, sebagai berikut (Bekkar et al., 2013).

$$\text{precision} = \frac{TP}{TP+FP} \times 100\% \dots\dots\dots(4)$$

2.8.4 F1 Score

F1 score adalah nilai rata-rata harmonik dari *recall* dan *precision* (Bekkar et al., 2013). *F1 score* dinyatakan dengan Persamaan 5, sebagai berikut.

$$\text{precision} = \frac{TP}{TP+\frac{1}{2}(FP+FN)} \times 100\% \dots\dots\dots(5)$$

2.9 Hyperparameter

Hyperparameter adalah konfigurasi yang berada diluar model. *Hyperparameter* membantu dalam menemukan parameter model yang tidak bergantung pada *data training* (Hutter, Lucke, & Schmidt-Thieme, 2015). *Hyperparameter* yang biasanya digunakan yaitu *epoch*, *batch size*, *learning rate* dan *optimizer* .

2.9.1 Epoch

Epoch merupakan *hyperparameter* yang menentukan berapa kali proses akan dilakukan dalam masa *training* dalam *neural network* (Wibawa, 2017).

2.9.2 Batch Size

Batch size merupakan jumlah *training sample* yang digunakan dalam satu *iteration*. *Batch size* digunakan dalam proses training untuk menentukan jumlah contoh *data training* dan merupakan salah satu *hyperparameter* terpenting (Rochmawati, 2021).

2.9.3 *Optimizer Adam*

Optimizer Adam merupakan algoritma stokastik berdasarkan perkiraan adaptif dari order rendah untuk memperbarui bobot secara iteratif yang didasarkan pada data training. Adam merupakan kombinasi antara RMSprop dan Stochastic. Adam cocok diterapkan pada permasalahan data dengan *gradient* yang menyebar (Wibawa, 2017).

2.9.4 *Learning Rate*

Learning rate merupakan salah satu *hyperparameter training* yang ditetapkan untuk menghitung nilai koreksi bobot pada waktu proses *training* (Rochmawati, 2021).

2.10 *Loss Function*

Loss function berfungsi untuk membandingkan nilai hasil dari prediksi dengan nilai sebenarnya (Golik, *et al.*, 2013). *Cross Entrophy* merupakan salah satu *loss function* yang digunakan dalam klasifikasi dalam kasus *multi class classification*. *Cross Entrophy* dinyatakan dalam Persamaan 6 sebagai berikut.

$$error = \sum_i E_t \dots\dots\dots (6)$$

Dimana E_t dapat dilihat pada Persamaan 7.

$$E_t = - \sum_i y_t \times \log o_t \dots\dots\dots (7)$$

Persamaan 7 terdapat beberapa variabel yaitu, y_t adalah kelas sebenarnya pada *timestep* ke-t, o_t adalah kelas prediksi pada *timestep* ke-t, dan E_t adalah *error* dari output o_t .

III. METODOLOGI PENELITIAN

3.1 Tempat dan Waktu Penelitian

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

3.1.1 Tempat Penelitian

Penelitian dilaksanakan di Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam (FMIPA) Universitas Lampung dan Ruang 204 Laboratorium UPT TIK Universitas Lampung.

3.1.2 Waktu Penelitian

Penelitian ini dilakukan pada bulan Desember 2021 di semester tujuh ganjil hingga bulan Juni 2022 di semester delapan genap.

3.2 Alat dan Bahan

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

3.2.1 Perangkat Keras (*Hardware*)

Laptop Acer Aspire Z3-451 dengan spesifikasi RAM 8.00 GB, *Harddisk* 500GB dan *Processor* AMD Richland A10-5757M

CPU 2.5 GHz sebagai alat pendukung penelitian untuk proses pengolahan citra kupu-kupu.

3.2.2 Perangkat Lunak (*Software*)

Perangkat lunak yang digunakan pada penelitian ini yaitu :

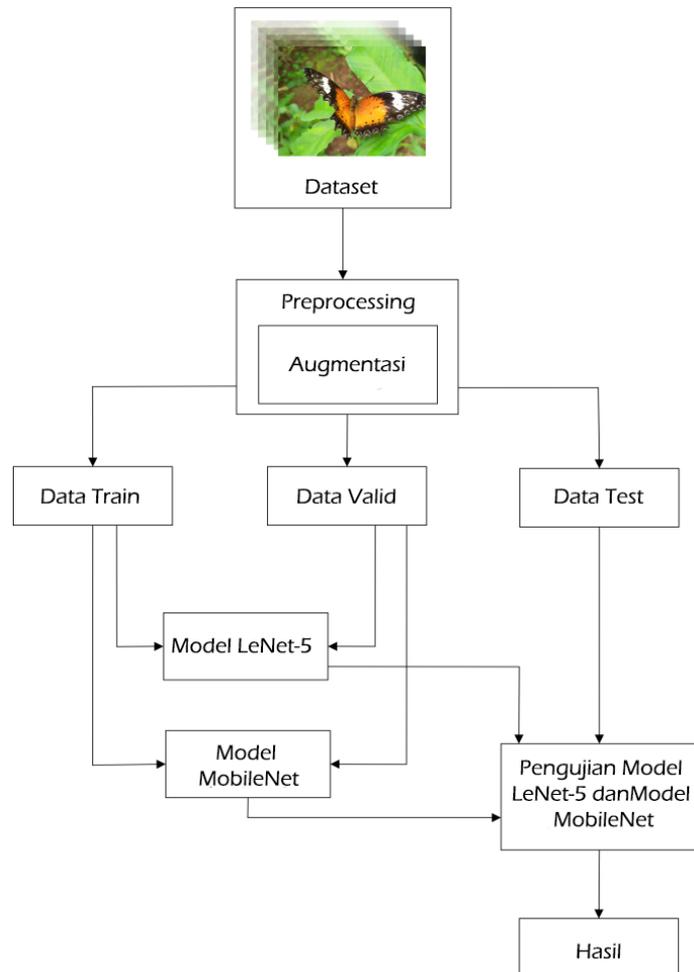
- a. Sistem Operasi *Windows 10 Pro 64-Bit* sebagai sistem operasi pada laptop.
- b. *Google Colaboratory*, *Jupyter Notebook* dan *Google Drive* sebagai pendukung perangkat lunak dalam proses identifikasi kupu-kupu.

3.2.3 Bahan Penelitian

Bahan yang digunakan pada penelitian ini berjumlah 800 citra kupu-kupu yang didapatkan dari tempat penangkaran kupu-kupu Taman Kupu-Kupu Gita Persada dan ditambah dengan teknik *augmentasi rotation* sehingga data menjadi 1600 citra, terdiri dari 200 citra Kupu Jarak (*Ariadne ariadne*), 200 citra Kupu Batik Cap (*Cethosia penthesilea*), 200 citra Kupu Harimau Kuning Hijau (*Parantica aspiasa*), 200 citra Kupu Pantat Merah (*Pacliopta aristolochiae*), 200 citra Kupu Raja Limau Jantan (*Papilio memnon*), 200 citra Kupu Jojo (*Papilio nephelus*), 200 citra Kupu Hijau Biru (*Papilio peranthus*), dan 200 citra Kupu Raja Helena (*Troides helena*).

3.3 Tahapan Penelitian

Tahapan penelitian yang digunakan dalam perbandingan arsitektur *deep learning LeNet* dan *MobileNet* dengan *augmentasi data* pada identifikasi kupu-kupu dapat dilihat pada Gambar 11.



Gambar 11. Tahapan 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 100 citra untuk masing-masing kelasnya dengan format JPG. *Dataset* diunduh dan

didalamnya terdapat 8 kelas kupu-kupu dengan total 800 citra. Jumlah *dataset* dapat dilihat pada Tabel 2.

Tabel 2. Jumlah *dataset*.

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

3.3.2 *Preprocessing*

Tahap *preprocessing* yang digunakan dalam penelitian ini yaitu *augmentasi* dengan teknik *rotation*. *Dataset* awal berjumlah 800 citra setelah dilakukan *augmentasi rotate* menghasilkan jumlah pada *dataset* bertambah menjadi 1600 citra dengan 8 kelas dan data gambar diubah ukurannya menjadi 224×224 piksel. Jumlah *dataset* setelah *augmentasi* dapat dilihat pada Tabel 3.

Tabel 3. Jumlah *dataset* setelah *augmentasi*.

Kelas	Jumlah Citra
Kupu Jarak	200
Kupu Batik Cap	200
Kupu Harimau Kuning Biru	200
Kupu Hijau Biru	200

Kelas	Jumlah Citra
Kupu Jojo	200
Kupu Pantat Merah	200
Kupu Raja Helena	200
Kupu Raja Limau	200
Total	1600

3.3.3 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 1120 citra kupu-kupu.

b. Data validasi

Data validasi atau data *validation* adalah data yang digunakan untuk melakukan proses validasi model dan mencegah *overfitting*. Data yang digunakan dalam penelitian ini untuk proses *validation*, yaitu 10% dengan total data 160 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 320 citra kupu-kupu.

3.3.4 Pelatihan Model

Tahap pelatihan model dalam penelitian ini menggunakan dua model yaitu *LeNet* dan *MobileNet* dengan data yang akan dilatih sebanyak 1600 citra kupu-kupu. Pelatihan data terhadap kedua

arsitektur ini menggunakan *tools google colab* dan *jupyter notebook* dengan *hyperparameter* yang sama, sehingga perbandingannya bisa terlihat. Pelatihan ini menggunakan *hyperparameter* yang dapat dilihat pada tabel 4.

Tabel 4. *Hyperparameters.*

Nama Parameter	Nilai
<i>Input sizes</i>	224×224×3
<i>Epoch</i>	10
<i>Batch size</i>	32
<i>Number of classes</i>	8
<i>Optimizer</i>	Adam
<i>Learning rate</i>	0,001

3.3.5 Pengujian *Model*

Tahap ini akan dilakukan pengujian terhadap data uji yang telah diperoleh dan hasil dari klasifikasi akan dicatat dan dilakukan analisis pada tahap selanjutnya.

3.3.6 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 *accuracy*, *recall*, *precision*, dan *f1 score*.

V. SIMPULAN DAN SARAN

5.1 Simpulan

Simpulan yang diperoleh berdasarkan hasil penelitian yang telah dilakukan adalah sebagai berikut :

1. Identifikasi kupu-kupu menggunakan arsitektur *LeNet* dan *MobileNet* berhasil dilakukan dengan tambahan *augmentasi data rotate*.
2. Hasil akurasi terbesar didapatkan oleh *MobileNet* menggunakan *augmentasi rotate* dengan mesin *tesla* dengan akurasi 99.37%.
3. Hasil analisis misklasifikasi antara arsitektur *LeNet* dan *MobileNet* terdapat 11 citra yang misklasifikasi yang terdapat pada kelas kupu raja helena, kupu pantat merah, kupu jarak, kupu harimau hijau kuning, dikarenakan adanya citra yang memiliki pola, warna dan posisi yang serupa.
4. Nilai ketepatan model dalam memprediksi kelas kupu-kupu (*precision*) dan nilai efektivitas pencarian data uji dalam menemukan kelas (*recall*) terbesar oleh arsitektur *MobileNet* menggunakan *augmentasi rotate* dengan mesin *tesla* memperoleh rata-rata nilai *recall* sebesar 98,12% dan rata-rata nilai *precision* sebesar 98,25%.
5. Jumlah parameter *LeNet* lebih banyak dibandingkan dengan arsitektur *MobileNet*, tetapi tingkat akurasi terbesar didapatkan oleh arsitektur *MobileNet* menggunakan *augmentasi rotate*, hal ini membuktikan efektivitas kinerja dari *MobileNet* dengan parameter yang lebih sedikit.

5.2 Saran

Saran yang diberikan untuk penelitian selanjutnya adalah sebagai berikut:

1. Menambahkan jumlah kelas dengan jenis kupu-kupu lainnya.
2. Mengembangkan klasifikasi kupu-kupu dengan *hyperparameter* yang dimodifikasi dari penelitian ini.
3. Mengembangkan klasifikasi kupu-kupu menggunakan model arsitektur yang lainnya seperti *Inception*, *DenseNet*, *ResNet*, *Xcaption* dan lain-lain.
4. Penelitian ini dapat dikembangkan dalam bentuk aplikasi berbasis *mobile* maupun *website*.

DAFTAR PUSTAKA

- Albawi, A., Mohammed, T., & A-Zawi, S. (2017). Understanding of a Convolutional Neural Network. *International Conference on Engineering*.
- Almryad, A. S., & Kutucu, H. (2020). Automatic Identification for Field Butterflies by Convolutional Neural Networks. *Engineering Science and Technology, an International Journal*.
- Alwanda, M., Ramadhan, R., & Alamsyah, D. (2020). Implementasi Metode Convolutional Neural Network Menggunakan Arsitektur LeNet-5 untuk Pengenalan Doodle. *Algoritme vol 1*.
- Andrian, R., Anwar, S., Muhammad, M., & Junaidi, A. (2019). Identifikasi Kupu-Kupu Menggunakan Ekstraksi Fitur Tepi (Edge Detection) dan Klasifikasi K-Nearest Neighbor (KNN). *Jurnal Teknik Informatika dan Sistem Informasi*.
- Andrian, R., Maharani, D., Muhammad, M., & Junaidi, A. (2019). Butterfly identification using gray level co-occurrence matrix (glcm) extraction feature and k-nearest neighbor (knn) classification. *Jurnal Ilmiah Teknologi Sistem Informasi*.
- Batubara, N., & Awangga, R. (2020). TUTORIAL Object Detection Plate Number With Convolutional Neural Network. *Joined Journa (Journal of Informatics Education)*.
- Bekkar, M., Djemaa, H. K., & Alitouche, T. (2013). Evaluation Measures for Models Assessment over Imbalanced Data Sets. *Journal of Information Engineering and Applications*.
- Borugadda, P., Lakshmi, R., & Govindu, S. (2021). Classification of Cotton Leaf. *Current Journal of Applied Science and Technology*.
- Eka Putra, W. S. (2016). Klasifikasi Citra Menggunakan Convolutional Neural Network (CNN) pada Caltech 101. *Jurnal Teknik ITS*, 5.

- Fitriati, D. (2016). Perbandingan Kinerja CNN LeNet 5 Dan Extreme Learning Machine Pada Pengenalan Citra Tulis Angka. *Jurnal Teknologi Terpadu*.
- Golik, P., Doetsch, P., & Ney, H. (2013). Cross-Entropy and Squared Error training: A theoretical and experimental comparison. *Proc. Annu. Conf. Int. Speech Commun. Assoc. INTERSPEECH*.
- Han, J., & Kamber, M. (2011). Data Mining : Concepts and Techniques. *Elsevier*.
- Hutter, F., Lucke, J., & Schmidt-Thieme, L. (2015). Beyond Manual Tuning of Hyperparameters. *KI*.
- Ilahiyah, S., & Nilogiri, A. (2018). Implementasi Deep Learning Pada Identifikasi Jenis Tumbuhan Berdasarkan Citra Daun Menggunakan Convolutional Neural Network. *JUSTINDO (Jurnal Sistem Dan Teknologi Informasi Indonesia)*.
- Koneri, R., & Saroyo, S. (2012). Distribusi dan keanekaragaman kupu-kupu (Lepidoptera) di Gunung Manado Tua, Kawasan Taman Nasional laut Bunaken, Sulawesi Utara.
- Krishna, S., & Kalluri, H. (2019). Deep Learning dan Transfer Learning Approaches for Image Classification. *Int. J. Recent Technol Eng*.
- Michael, & Hartati, E. (2022). Klasifikasi Spesies Kupu-Kupu Menggunakan Metode Convolutional Neural Network. *MDP Student Conference*.
- Pan, H., Pang, Z., Wang, Y., & Chen, L. (2017). A new image recognition and classification method combining Transfer Learning Algorithm and MobileNet model for welding defects. *IEEE*.
- Pangestu, R. A., Rahmat, B., & Anggraeny, F. T. (2020). Implementasi algoritma CNN untuk klasifikasi citra lahan dan perhitungan luas. *Jurnal Informatika dan Sistem Informasi (JIFoSI)*, 1(1), 166 - 174.
- Rochmawati, N. (2021). Analisa Learning Rate dan Batch Size pada Klasifikasi Covid Menggunakan Deep Learning dengan Optimizer Adam. *Journal Information Engineering and Education Technology*.
- Shorten, C., & Khosgoftaar, T. (2019). A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data*.
- Sihombing, H., Buulolo, I., Toba, K., & Utara, S. (2021). Pengenalan Buah Kopi Berdasarkan Parameter Warna Menggunakan Algoritma Backpropagation dan Algoritma Support Vector Mechine (SVM). *Seminastika*.

- Soekardi. (2007). *Kupu-kupu di Kampus Unila*. Bandar Lampung: Universitas Lampung.
- Soekardi, H., Alia, L., Anshori, D., & Martinus. (2016). *Kupu-Kupu Lampung Taman Kupu-Kupu Gita Persada Bandar Lampung*. Yayasan Sahabat Alam.
- Suyanto. (2018). *Machine Learning Tingkat Dasar dan Lanjut*. Bandung: Informatika Bandung.
- Takahashi, R., Matsubara, T., Member, IEEE, & Uchara, K. (2015). Data Augmentasi using Random Image Cropping and Patching for Deep CNNs. *Journal of LATEX class Files*.
- Venkateswarlu, I., Kakarla, J., & Prakash, S. (2020). Face mask detection using MobileNet and Global Pooling Block. *IEEE*.
- W, I. H., Jondri, & Rizal, A. (2021). Klasifikasi Suara Paru-Paru Menggunakan Convolutional Neural Network (CNN). *e-Proceeding of Engineering*, 8, 3218-3223.
- Wibawa, M. (2017). Pengaruh Fungsi Aktivasi, Optimisasi dan Jumlah Epoch Terhadap Performa Jaringan Syaraf Tiruan. *Jurnal Sistem Dan Informatika*.
- Yuzkat, M., Ilhan, H. O., & Aydin, N. (2021). Multi-model CNN fusion for sperm morphology analysis . *Computers in Biology and Medicine* .