

Sistem Klasifikasi Jenis Kupu-Kupu Menggunakan Visual Geometry Group 16

Yusup Miftahuddin ¹, Farhan Adani ²

^{1,2} Program Studi Informatika, Institut Teknologi Nasional Bandung

Email : yusufm@itenas.ac.id

Received DD MM YYYY | Revised DD MM YYYY | Accepted DD MM YYYY

ABSTRAK

Kupu-kupu merupakan serangga yang memiliki corak maupun warna bervariasi yang membedakan dengan jenis kupu-kupu lainnya. Berbeda dengan para ahli di bidang fauna khususnya kupu-kupu, untuk orang awam tentu saja sedikit sulit untuk membedakan setiap jenis kupu-kupu yang beraneka ragam jenisnya. Metode yang seringkali digunakan dalam mengenali jenis kupu-kupu yaitu secara manual dengan indera penglihatan saja, tentu cara tersebut tidak cukup akurat atau rentan akan kesalahan. Oleh karena itu pada penelitian ini kami mengusulkan algoritma Convolutional Neural Network dengan arsitektur Visual Geometry Group 16 atau VGG-16 untuk mengklasifikasi 10 jenis kupu-kupu. Hasil yang diperoleh dari penelitian yang telah dilakukan dengan menerapkan beberapa parameter seperti penggunaan optimizer Adam dan epoch sebanyak 100 menghasilkan nilai akurasi terbaik sebesar 89%.

Kata kunci: Kupu-kupu, Convolutional Neural Network, VGG-16.

ABSTRACT

Butterfly is insect that have different variety pattern and color with the other kind of butterfly. Different with animal experts especially butterfly, certainly little bit difficult for common people to differentiate any kind of butterfly. The method that often used to recognized kind of butterfly just manually by sight senses, of course that way is not quite accurate or vulnerable to errors. Therefore on this research we have proposed an algorithm of convolutional neural network with visual geometry group 16 architecture or vgg-16 to classify 10 kinds of butterflies . The results of this research has been done by applying some parameters like using adam optimizer and 100 of epochs produced the best accuracy with a value 89 %.

Keywords: Butterfly, Convolutional Neural Network, VGG-16.

1. PENDAHULUAN

KUPU-KUPU ADALAH SERANGGA YANG TERMASUK ORDO LEPIDOPTERA. DI INDONESIA SENDIRI DIPERKIRAKAN TERDAPAT 2.000 LEBIH JENIS KUPU-KUPU DARI TOTAL PERKIRAAN 17.000 LEBIH KUPU-KUPU DI DUNIA. DARI TOTAL PERKIRAAN JENIS KUPU-KUPU DI INDONESIA TERSEBUT, HANYA SEKITAR SETENGAHNYA SAJA YANG SUDAH DIKETAHUI JENIS NYA (**ARRUMMAISHA ET AL., 2014**). KUPU-KUPU MEMILIKI BERBAGAI CORAK DAN WARNA, CORAK DAN WARNA PADA KUPU-KUPU INILAH YANG MEMBEDAKAN ANTARA JENIS LAINNYA.

PENGENALAN ATAUPUN PENGKLASIFIKASIAN JENIS KUPU-KUPU DIBUTUHKAN APABILA INGIN MEMISAHKAN KUPU-KUPU KE MASING-MASING JENIS NYA. METODE YANG BIASA DIGUNAKAN UNTUK MEMBEDAKAN JENIS KUPU-KUPU CONTOHNYA PADA TAMAN KUPU-KUPU DI LAMPUNG HANYA DENGAN BERDASARKAN PENGLIHATAN MANUSIA SAJA (**ANDRIAN ET AL., 2019**). TENTU SAJA CARA SEPERTI ITU TIDAK CUKUP BAIK DALAM MEMBEDAKAN JENIS KUPU-KUPU DAN RENTAN AKAN KESALAHAN. PENGIDENTIFIKASIAN JENIS KUPU-KUPU MEMILIKI AKURASI RENDAH DAN PENGENALAN NYA CUKUP LAMBAT DIKARENAKAN JENIS YANG BERANEKA RAGAM DAN JUGA MEMILIKI KEMIRIPAN CUKUP TINGGI ANTARA SETIAP JENISNYA (**ALMRYAD & KUTUCU, 2020**). UNTUK ORANG AWAM PASTI CUKUP SULIT DALAM MEMBEDAKAN JENIS KUPU-KUPU. BERDASARKAN PERMASALAHAN TERSEBUT MAKA PADA PENELITIAN INI DIBUAT SISTEM UNTUK MENGLASIFIKASI JENIS KUPU-KUPU.

PENELITIAN YANG AKAN DILAKUKAN ADALAH MENGLASIFIKASIKAN JENIS DARI KUPU-KUPU MENGGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK DENGAN ARSITEKTUR VISUAL GEOMETRY GROUP (VGG) 16 SEBAGAI SALAH SATU JENIS MODEL DARI CNN. ARSITEKTUR VGG-16 MEMILIKI 13 CONVOLUTION *LAYER* DAN 3 FULLY CONNECTED SEHINGGA MEMILIKI TOTAL 16 *LAYER* (**SIMONYAN & ZISSERMAN, 2015**). VGG-16 JUGA MERUPAKAN MODEL PRE-TRAINED DARI CNN YANG MEMANFAATKAN CONVOLUTIONAL *LAYER* DENGAN CONVOLUTIONAL FILTER 3X3. UKURAN TERSEBUT TERMASUK KE DALAM UKURAN YANG KECIL SEHINGGA KEDALAMAN NEURAL NETWORK DAPAT DITAMBAH CONVOLUTIONAL *LAYER* LAGI, MAKA AKAN MENGHASILKAN MODEL CNN MENJADI LEBIH AKURAT DIBANDINGKAN MODEL CNN LAINNYA (**PANGESTU & BUNYAMIN, 2018**).

PENELITIAN SEBELUMNYA YANG MENGGUNAKAN METODE JARINGAN SYARAF TIRUAN ATAU CONVOLUTIONAL NEURAL NETWORK DENGAN ARSITEKTUR VISUAL GEOMETRY GROUP (VGG) 16 TELAH BANYAK DILAKUKAN SALAH SATU CONTOHNYA PADA PENELITIAN (**SETIAWAN, 2020**) DENGAN JUDUL PERBANDINGAN ARSITEKTUR CONVOLUTIONAL NEURAL NETWORK UNTUK KLASIFIKASI FUNDUS. PADA PENELITIAN TERSEBUT MEMBANDINGKAN 9 ARSITEKTUR CNN SALAH SATUNYA VGG-16. HASIL AKURASI MODEL TERBAIK YANG DIDAPAT ADALAH VGG16 DENGAN AKURASI SEBESAR 92,31% DENGAN SENSITIVITAS 90,7%, DAN SPESIFITAS 94%. MAKA DARI PENJELASAN TERSEBUT PENELITIAN INI DILAKUKAN UNTUK MENGUKUR TINGKAT AKURASI METODE YANG DIGUNAKAN CONVOLUTIONAL NEURAL NETWORK DENGAN ARSITEKTUR VISUAL GEOMETRY GROUP (VGG) 16 DALAM MENGLASIFIKASI JENIS KUPU-KUPU.

2. TEORI

2.1. KUPU-KUPU

Kupu-kupu merupakan serangga yang tergolong ke dalam ordo lepidoptera, atau serangga yang memiliki sisik. Berbagai macam warna pada kupu-kupu menjadi salah satu hal penting untuk mengidentifikasi jenis dari kupu-kupu (**Arrummaisha et al., 2014**). Ciri-ciri dari kupu-

kupu diantaranya memiliki 3 pasang kaki pada bagian dada, permukaan tubuh tersusun atas zat kitin, merupakan hewan yang aktif di siang maupun malam hari dan memiliki 2 pasang sayap yang cukup indah.

2.2. CONVOLUTIONAL NEURAL NETWORK

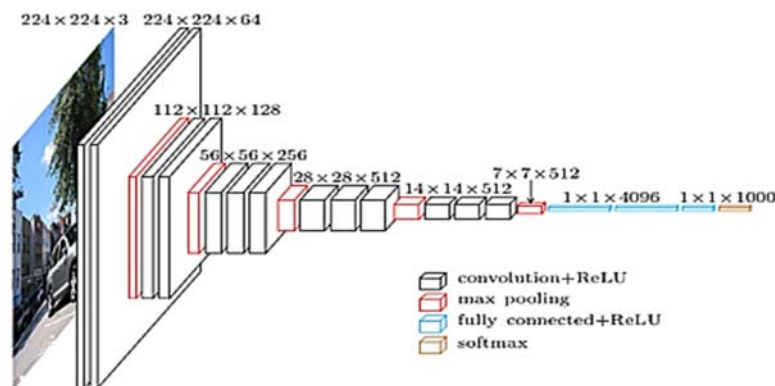
Convolutional Neural Network (CNN) merupakan jenis *deep learning*. Pada CNN dilakukan proses operasi konvolusi dengan menggabungkan beberapa elemen pemrosesan secara paralel yang terinspirasi dari sistem saraf biologis makhluk hidup. CNN merupakan pengembangan *multilayer* perceptron yang dibangun untuk pengolahan data 2 dimensi. CNN memiliki struktur yang terdiri dari input, ekstraksi fitur, klasifikasi dan *output*. Lapisan yang terdiri pada ekstraksi fitur adalah lapisan tersembunyi atau *hidden layer*, *convolutional layer*, fungsi aktivasi seperti ReLU dan *pooling layer*. Ekstraksi fitur dilakukan untuk memperoleh fitur yaitu sebuah tekstur dari citra biner, RGB maupun grayscale. Terdapat beberapa metode dalam ekstraksi fitur seperti metode statis, struktural dan spektral [9]. Kemudian pada proses klasifikasi terdiri dari *flatten*, fungsi aktivasi dan *fully-connected layer*. CNN banyak dimanfaatkan dalam aplikasi untuk pemrosesan citra atau gambar (Hu et al., 2015).

2.3. TRANSFER LEARNING

Transfer learning adalah bagian dari machine learning yang berfungsi untuk mengenal objek suatu citra. *Transfer learning* memanfaatkan model yang sudah dilatih sebelumnya pada suatu *dataset*. *Transfer learning* sering disebut juga sebagai Pre-trained Network. Transfer learning juga dapat digunakan lebih cepat dan juga mudah daripada harus melatih jaringan yang baru dari awal. Keuntungan dari transfer learning adalah jaringan pre-trained sudah mempelajari terlebih dahulu beberapa fitur. Sehingga fitur ini dapat diterapkan dan dimanfaatkan. Pada CNN dibutuhkan *dataset* yang cukup besar, oleh karena itu pemanfaatan transfer learning ini dapat digunakan sebagai opsi untuk melatih suatu data (Tan et al., 2018).

2.4. VISUAL GEOMETRY GROUP 16

Visual Geometry Group 16 (VGG-16) merupakan salah satu arsitektur dari CNN yang digunakan dalam kompetisi ISL VR (ImageNet) di tahun 2014. Arsitektur VGG-16 memiliki 13 Convolution *layer* dan 3 fully connected sehingga memiliki total 16 *layer* (Pangestu & Bunyamin, 2018). *Layer* Convolutional pada VGG-16 memiliki filter 3x3 dan max pooling *layer* 2x2. VGG-16 banyak di implementasi kan dalam teknik klasifikasi suatu citra dan cukup populer karena mudah untuk diimplementasikan. Arsitektur VGG-16 merupakan peningkatan arsitektur AlexNet dengan memodifikasi dan mengganti filter besar dengan filter berukuran 3x3. Arsitektur VGG-16 digambarkan seperti gambar berikut:



Gambar 1. Arsitektur VGG-16

Pada Gambar diatas seluruh lapisan konvolusi memiliki ukuran kernel 3x3. Untuk lapisan konvolusi 1 & 2 memiliki jumlah filter 64, lapisan 3 dan 4 konvolusi memiliki 128 filter, lapisan 5,6 dan 7 memiliki 256 filter, lapisan 8 sampai 13 memiliki 512 filter. kemudian untuk proses *max-pooling* 2 x 2 dilakukan setelah lapisan 2,4,7,10 dan 13. Pada *max-pooling* lapisan terakhir akan terhubung dengan lapisan fully connected dan akan terhubung ke classifier dalam penentuan *output* ataupun kelas dari citra yang diuji. Berikut merupakan penjelasan setiap lapisan pada VGG-16:

a. *Convolutional Layer*

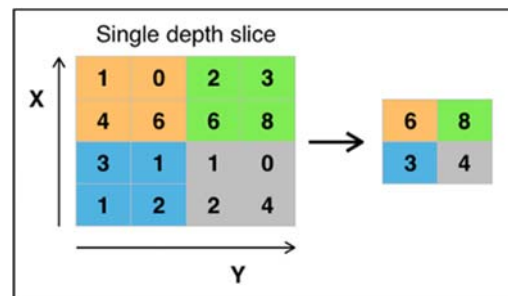
Pada lapisan ini dilakukan ekstraksi objek dari citra input dengan menggunakan filter. *Layer* ini juga merupakan *layer* pertama yang menerima inputan citra atau gambar pada arsitektur. *Convolutional layers* terdiri dari beberapa neuron sehingga membentuk suatu filter. Filter ini berisi nilai atau bobot yang digunakan untuk melakukan proses selanjutnya seperti mendeteksi objek sehingga menghasilkan karakter yang dapat dikenal dari tepi ataupun warna dari objek tersebut.

b. Fungsi aktivasi ReLU

Fungsi aktivasi merupakan fungsi non linear untuk menyelesaikan permasalahan non-trivial pada sebuah jaringan (**Arrofiqoh & Harintaka, 2018**). Fungsi aktivasi pada CNN berada pada perhitungan terakhir atau feature map. Rectified Linear Unit (ReLU) merupakan salah satu operasi nonlinear untuk menghilangkan *vanishing gradient* dan meningkatkan representasi dari model.

- *Pooling Layer*

Pooling layer merupakan lapisan yang mengelola fungsi *Feature map* ke dalam berbagai macam operasi berdasarkan nilai piksel paling dekat. Bentuk umum lapisan Pooling adalah filter berukuran 2x2. *Pooling layers* berfungsi untuk mengontrol terjadinya *overfitting* dan juga untuk mempercepat komputasi dengan cara mereduksi jumlah parameter dalam jaringan. *Pooling layer* sering disebut juga sub sampling. Jenis pooling yang sering digunakan yaitu average pooling dan *Max pooling* (**Bejiga et al., 2017**). Berikut merupakan ilustrasi dari salah satu jenis *pooling* yaitu *Max pooling*:



Gambar 2. *Max pooling*

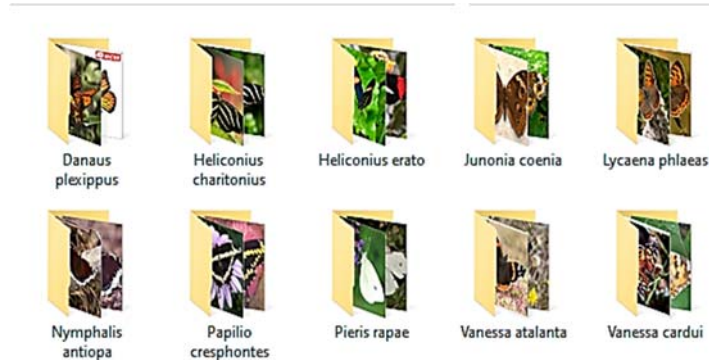
- Fully-connected *layer*

Lapisan fully-connected merupakan lapisan yang terdiri dari input *layer*, hidden *layer* dan *output layer*. Fungsi dari lapisan ini adalah menghubungkan semua node menjadi 1 dimensi (**Albelwi & Mahmood, 2017**). Pada lapisan ini neuron aktivasi yang berada pada *layer* sebelumnya bergabung dengan neuron pada *layer* berikutnya. Pada *layer* ini menerima input dari hasil *output layer* ekstraksi fitur dan kemudian hasil dari *output* tersebut merupakan nilai untuk proses klasifikasi (**Cholissodin & Soebroto, 2020**).

3. HASIL

3.1. PENGGUNAAN DATASET

Dataset yang digunakan merupakan *dataset* Leeds Butterfly yang diperoleh dari Kaggle dan tersedia untuk umum. Pada penelitian ini dideteksi 10 jenis kupu-kupu. dengan jumlah gambar sebanyak 2.088, yaitu kupu-kupu jenis *Danaus plexippus*, *Heliconius charitonius*, *Heliconius erato*, *Junonia coenia*, *Lycaena phlaeas*, *Nymphalis antiopa*, *Papilio cresphontes*, *Pieris rapae*, *Vanessa atalanta* dan *Vanessa cardui*. Berikut merupakan gambar dari sepuluh kelas data yang akan diuji.



Gambar 3. Pembagian *Dataset*

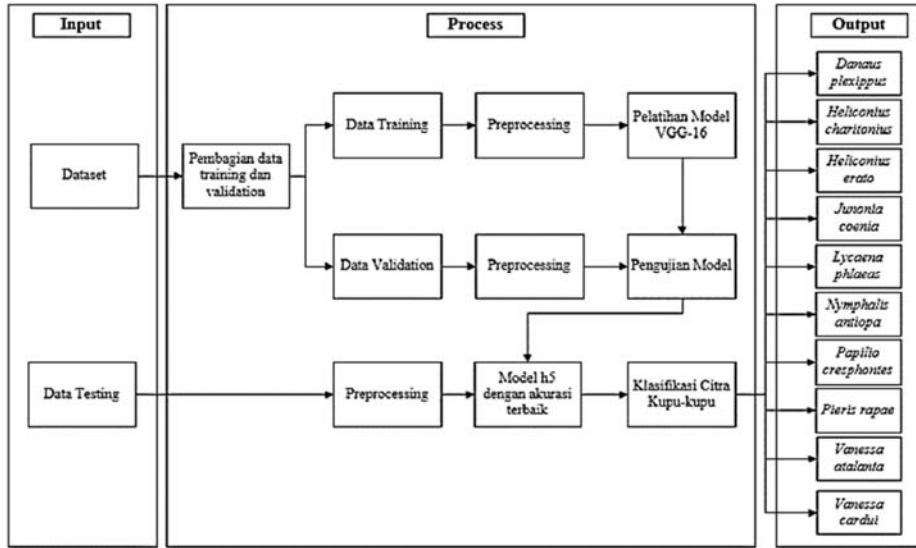
Citra kupu-kupu yang diperoleh memiliki ukuran citra yang berbeda dengan channel warna RGB (red, green dan blue). Dengan jumlah *dataset* di atas pembagian *dataset* untuk data *training*, *testing* dan *validation* dari masing kelas adalah sebagai berikut :

Tabel 1. Pembagian *Dataset*

Kelas	Data <i>Training</i>	Data <i>Testing</i>	Data <i>Validation</i>
Danaus plexippus	174	23	21
Heliconius charitonius	228	29	28
Heliconius erato	90	12	11
Junonia coenia	148	19	18
Lycaena phlaeas	163	21	20
Nymphalis antiopa	180	24	22
Papilio cresphontes	115	15	14
Pieris rapae	177	23	22
Vanessa atalanta	191	25	23
Vanessa cardui	201	26	25

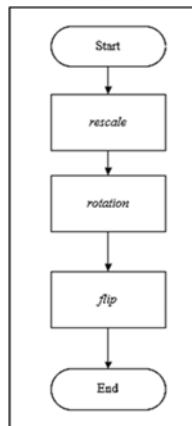
Dari sekitar 2.088 total citra pada *dataset*, dibagi menjadi 3 bagian yaitu data *training*, *testing* dan *validation*. Pembagian *dataset* tersebut berguna untuk proses pelatihan dan pengujian model terhadap *dataset* yang digunakan.

3.2. PERANCANGAN SISTEM



Gambar 4. Blok Diagram Sistem

Tahap pertama adalah mempersiapkan *dataset* dari citra yang akan dijadikan sebagai objek penelitian. Dalam penelitian ini objek yang digunakan adalah citra kupu-kupu dari 10 kelas jenis yang berbeda. *Dataset* tersebut dibagi menjadi data *training*, *validation* dan *testing* dengan perbandingan 80:10:10 untuk melakukan pengujian model. Sehingga dari total 2.088 data dibagi menjadi 1.667 data *training*, 204 data *validation* dan 217 data *testing*. Dalam kasus ini model yang digunakan adalah arsitektur Visual Geometry Group 16 (VGG-16). Data yang telah dibagi menjadi data *training* dan data *validation* tersebut kemudian dilakukan pre-processing seperti *rescale*, *rotation*, flip dan *zoom* dan sebagainya.

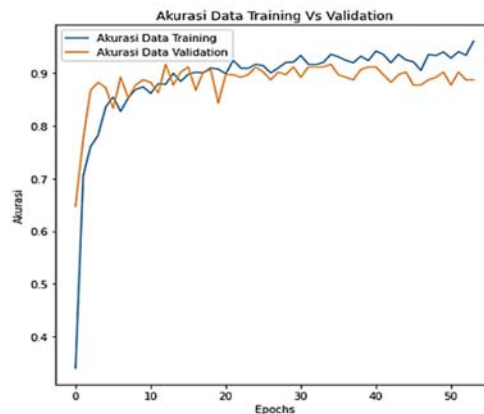


Gambar 5. Flowchart *preprocessing*

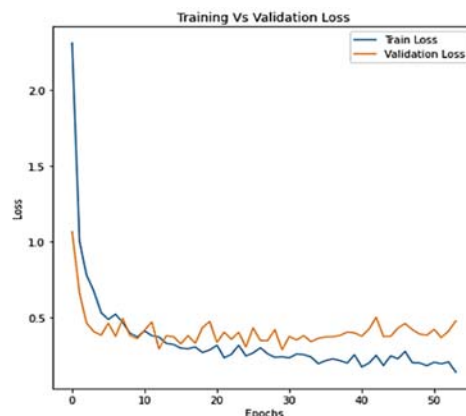
Pada tahap ini dilakukan proses *Data augmentation*. *Data augmentation* yaitu proses meningkatkan ukuran set latihan agar mendapatkan banyak gambar yang berbeda. Tahap ini juga merupakan cara lain untuk mengurangi *overfitting* sebuah model dengan cara meningkatkan jumlah data pelatihan menggunakan informasi suatu data. Data yang berupa citra akan dilakukan proses seperti *rescale*, *rotation*, flip transformasi dan Setelah melakukan pre-processing langkah selanjutnya adalah pelatihan terhadap model yang digunakan yaitu VGG-16, dimana bobot yang diambil dari model VGG-16 ini sudah dilatih pada ImageNet. Pada

tahapan ini proses yang dilakukan yaitu konvolusi, fungsi aktivasi ReLU, *Max pooling* dan SoftMax. Setelah melakukan pelatihan terhadap model yang sudah dilatih kemudian model diuji untuk menghasilkan model yang terbaik. Ketika melakukan pengujian, model yang diambil adalah model yang memiliki tingkat akurasi yang tinggi dan model akan disimpan untuk melakukan pengujian pada data *testing*. Hasil akhir berupa fitur dari model yang sudah diperoleh sebelumnya dievaluasi pada data *testing*. Setelah fitur dalam citra diperoleh maka akan diklasifikasikan untuk menghasilkan *output* berupa 10 kelas dari citra kupu-kupu yaitu *Danaus plexippus*, *Heliconius charitonius*, *Heliconius erato*, *Junonia coenia*, *Lycaena phlaeas*, *Nymphalis antiopa*, *Papilio cresphontes*, *Pieris rapae*, *Vanessa atalanta* dan *Vanessa cardui*. Pengujian model dilakukan untuk menguji model yang di diterapkan pada penelitian ini yaitu model Visual Geometry Group 16 (VGG-16).

Model tersebut diuji terhadap data yang telah dibagi menjadi data *training*, *testing* dan *validation*. Data validasi digunakan untuk mengukur akurasi *training* pada setiap epoch. Model tersebut diuji terhadap data *testing* yang sudah dibuat sebelumnya terdapat sebanyak 1667 data atau gambar dari 10 jenis dari kupu-kupu *Danaus plexippus*, *Heliconius charitonius*, *Heliconius erato*, *Junonia coenia*, *Lycaena phlaeas*, *Nymphalis antiopa*, *Papilio cresphontes*, *Pieris rapae*, *Vanessa atalanta* dan *Vanessa cardui*. Akurasi yang diperoleh pada saat proses *training* sebesar 89%. Pengukuran akurasi pada pengklasifikasian jenis kupu-kupu ini dilihat dari nilai probabilitas yang diperoleh dari tiap kelas. Akurasi model terhadap citra yang diuji akan baik apabila nilai probabilitas mendekati nilai 1 dimana nilai probabilitas ini terdiri dari rentang 0 hingga 1.



Gambar 6. Grafik akurasi *training* dan *validation*



Gambar 7. Grafik *loss training* dan *validation*

Gambar diatas merupakan grafik dari hasil akurasi dan loss data *training* dan validasi pada saat pembuatan model. Akurasi merupakan rasio prediksi benar baik itu positif maupun negatif dari seluruh data tiap kelas yang ada. Sedangkan dari *loss function* model dapat mengetahui apakah prediksi sudah tepat atau belum. Dalam pembuatan sebuah model dilakukan dengan tujuan agar nilai loss nya rendah.

Dalam pembuatan model ini dilakukan pengukuran nilai loss dengan menggunakan cross entropy loss karena umum digunakan dalam proses klasifikasi. Proses pembuatan model dibangun dengan beberapa parameter lainnya diantaranya epoch = 100, batch size = 32, optimizer = Adam dan fungsi aktivasi ReLU. Waktu yang digunakan dalam membuat model VGG-16 ini adalah 825,7 Menit atau sekitar 13,7 Jam. Kurang lebih memakan waktu 950 detik atau 15 menit dalam pelatihan tiap satu epoch.

Setelah melakukan proses *training* selanjutnya model diuji atau dievaluasi terhadap data *testing*. Proses evaluasi ini mengukur beberapa nilai seperti nilai *Precision*, *Recall*, dan *F-Score*. *Precision* merepresentasikan pengukuran prediksi positif dengan seluruh nilai atau hasil yang diprediksi benar atau positif. *Recall* atau sensitivitas merupakan rasio yang diprediksi benar (positif) dengan seluruh data yang benar dari tiap kelas. Sedangkan *F-Score* adalah perbandingan nilai rata-rata antara *Precision* dan *Recall*. Tabel berikut merupakan hasil evaluasi model yang telah dilatih dengan data *testing*.

Tabel 2. Evaluasi Model

Kelas	<i>Precision</i>	<i>Recall</i>	<i>F-Score</i>
Danaus plexippus	0.724138	0.913043	0.807692
Heliconius charitonius	1.000000	0.965517	0.982456
Heliconius erato	0.916667	0.916667	0.916667
Junonia coenia	1.000000	0.947368	0.972973
Lycaena phlaeas	1.000000	0.952381	0.975610
Nymphalis antiopa	1.000000	0.958333	0.978723
Papilio cresphontes	0.875000	0.933333	0.903226
Pieris rapae	0.958333	1.000000	0.978723
Vanessa atalanta	0.916667	0.880000	0.897959
Vanessa cardui	1.000000	0.884615	0.938776

Confusion matrix merupakan metode dalam bentuk visualisasi yang digunakan untuk mengukur tingkat keberhasilan atau performa model dalam proses klasifikasi dalam machine learning. Gambar berikut merupakan representasi *confusion matrix* dari performa model yang di lakukan evaluasi terhadap data uji dengan menampilkan data aktual dan data prediksi yang dilakukan oleh model.

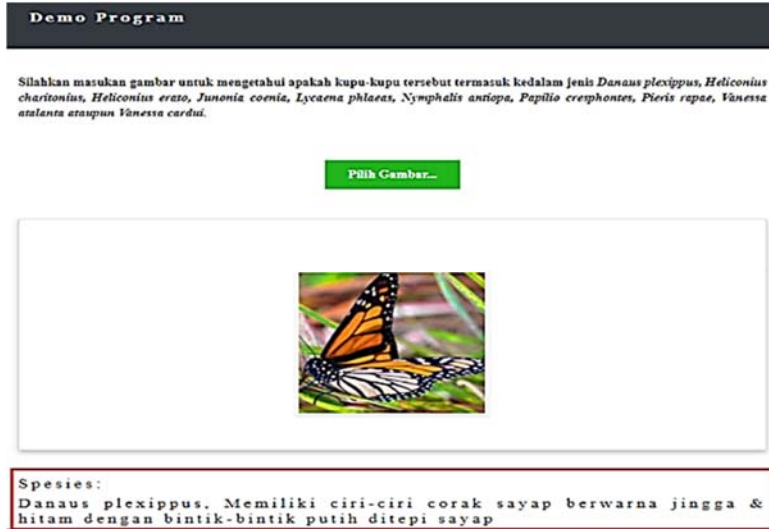
Confusion matrix

	Danaus plexippus	Heliconius charitonius	Heliconius erato	Junonia coenia	Lycæna phlaeas	Nymphalis antiopa	Papilio cressphontes	Pieris rapae	Vanessa atalanta	Vanessa cardui
Danaus plexippus	21	1	0	0	0	0	1	0	0	0
Heliconius charitonius	0	29	0	0	0	0	0	0	0	0
Heliconius erato	0	0	12	0	0	0	0	0	0	0
Junonia coenia	0	0	0	19	0	0	0	0	0	0
Lycæna phlaeas	0	0	0	0	17	0	0	0	0	4
Nymphalis antiopa	0	0	0	0	0	22	2	0	0	0
Papilio cressphontes	1	1	0	0	0	0	13	0	0	0
Pieris rapae	0	0	1	0	0	0	0	22	0	0
Vanessa atalanta	1	2	0	0	0	1	2	0	19	0
Vanessa cardui	2	0	0	0	0	1	2	0	0	21
	Danaus plexippus	Heliconius charitonius	Heliconius erato	Junonia coenia	Lycæna phlaeas	Nymphalis antiopa	Papilio cressphontes	Pieris rapae	Vanessa atalanta	Vanessa cardui
	Predicted									

Gambar 8. Confusion matrix

Pada gambar diatas dapat dilihat bahwa hasil prediksi yang diperoleh model dan dievaluasi terhadap setiap kelas yang ada pada data uji mendapat hasil yang cukup baik. Untuk kelas pertama yaitu Danaus plexippus terdapat 21 prediksi benar dari 23 data uji dimana terdapat 2 kesalahan prediksi yang memprediksi kelas Heliconius charitonius dan Papilio cressphontes. Untuk kelas Heliconius charitonius model mampu memprediksi kelas dengan benar sebanyak 29 data. Pada kelas Heliconius erato memperoleh hasil prediksi benar terhadap data uji yaitu sebanyak 12 data. Kelas Junonia coenia memperoleh prediksi benar sebanyak 19 data. Tidak ada kesalahan prediksi yang dilakukan oleh model. Untuk kelas Nymphalis antiopa kelas yang berhasil diprediksi sebanyak 22 dari total 24 gambar. Model memprediksi 2 kelas yang berbeda yaitu Papilio cressphontes. Kelas Papilio cressphontes mendapat hasil prediksi benar sebanyak 13 data. Namun prediksi salah pada kelas ini cukup merata yaitu 1 memprediksi Danaus plexippus, dan 1 memprediksi charitonius.

Pada kelas Pieris rapae model berhasil memprediksi sebanyak 22 data benar dan 1 data salah yang memprediksi kelas Heliconius erato. untuk 2 kelas terakhir yaitu Vanessa atalanta dan Vanessa cardui model mampu memprediksi 19 & 21 kelas pada masing-masing kelas. Total data uji pada Vanessa atalanta adalah 25 data dengan kesalahan prediksi sebanyak 6 data yang memprediksi kelas lain. Sedangkan untuk kelas Vanessa cardui model melakukan kesalahan prediksi sebanyak 5 data dengan memprediksi 2 data Danaus plexippus, 1 data Nymphalis antiopa dan 2 data Papilio cressphontes. Setelah tahapan implementasi hingga evaluasi model terhadap data uji telah selesai, pada penelitian ini juga mengimplementasikan model kedalam sebuah website berbasis Flash sebagai media agar memudahkan user melakukan pengujian untuk mengetahui jenis dari kupu-kupu yang ingin diketahui. Berikut merupakan gambarannya.



Gambar 9. Tampilan GUI

Pada gambar diatas menghasilkan tujuan dari penelitian ini yaitu menampilkan *output* berupa jenis dari kupu-kupu yang ingin diketahui jenisnya dan terdapat beberapa deskripsi dari jenis kupu-kupu itu sendiri. Apabila ingin mengetahui jenis lainnya pengguna hanya perlu memilih gambar lain dengan menekan tombol pilih gambar setelah itu sistem akan memprediksi termasuk ke dalam jenis apakah gambar kupu-kupu yang diuji. Terdapat 10 jenis kupu-kupu yang dapat diuji sesuai *dataset* yang digunakan diantaranya *Danaus plexippus*, *Heliconius charitonius*, *Heliconius erato*, *Junonia coenia*, *Lycaena phlaeas*, *Nymphalis antiopa*, *Papilio cresphontes*, *Pieris rapae*, *Vanessa atalanta* dan *Vanessa cardui*.

4. KESIMPULAN

Pada penelitian ini telah di implementasikan algoritma Convolutional Neural Network dengan arsitektur Visual Geometry Group 16 atau VGG-16. Pengimplementasian metode VGG-16 tersebut telah berhasil memberikan hasil yang cukup baik yaitu dengan tujuan untuk mengklasifikasi ataupun memprediksi berupa 10 kelas kupu-kupu dengan jenis berbeda yang menghasilkan *output* berupa nama jenis yang diuji dan deskripsi mengenai jenis kupu-kupu tersebut. Dari hasil penelitian yang telah dilakukan penerapan model VGG-16 yang telah dievaluasi terhadap data *testing* dengan pengaturan beberapa parameter seperti epoch 100, batch size 32, optimizer Adam mendapatkan hasil dengan akurasi sebesar 0,8986 atau 89% dan *loss function* 0,365. Data *testing* sebesar 10% hampir seluruh kelas nya memiliki kecocokan dengan data *training*. Tentu saja dalam proses pelatihan banyak aspek yang mempengaruhi hasil akurasi yang diperoleh seperti posisi atau ukuran dari gambar, pengaturan beberapa parameter seperti epoch ataupun optimizer yang tepat. Diharapkan penelitian lebih lanjut dapat menggunakan beberapa variasi algoritma CNN lainnya sehingga menghasilkan akurasi yang lebih baik lagi.

DAFTAR PUSTAKA

- Albelwi, S., & Mahmood, A. (2017). A framework for designing the architectures of deep Convolutional Neural Networks. *Entropy*, 19(6). <https://doi.org/10.3390/e19060242>
- Almryad, A. S., & Kutucu, H. (2020). Automatic identification for field butterflies by convolutional neural networks. *Engineering Science and Technology, an International*

- Journal*, xxx, 4–10. <https://doi.org/10.1016/j.jestch.2020.01.006>
- Andrian, R., Anwar, S., Muhammad, M. A., & Junaidi, A. (2019). Identifikasi Kupu-Kupu Menggunakan Ekstraksi Fitur Deteksi Tepi (Edge Detection) dan Klasifikasi K-Nearest Neighbor (KNN). *Jurnal Teknik Informatika Dan Sistem Informasi*, 5(2), 234–243. <https://doi.org/10.28932/jutisi.v5i2.1744>
- Arrofiqoh, E. N., & Harintaka. (2018). IMPLEMENTASI METODE CONVOLUTIONAL NEURAL NETWORK UNTUK KLASIFIKASI TANAMAN PADA CITRA RESOLUSI TINGGI (The Implementation of Convolutional Neural Network Method for Agricultural Plant Classification in High Resolution Imagery). *Geomatika*, 24(2), 61–68.
- Arrummaisha, L. D., Rahayu, S. E., & Sulisetijono. (2014). Preferensi kupu-kupu familia Nymphalidae dan Lycaenidae pada tumbuhan di wisata air terjun Coban Rais kota Batu Jawa Timur. *Jurnal Online UM*, 1(1), 1–7. <http://jurnal-online.um.ac.id/data/artikel/artikelAC6A3530FFDA5CA7E7EE0FB2EE6819DF.pdf>
- Bejiga, M. B., Zeggada, A., Nouffidj, A., & Melgani, F. (2017). A convolutional neural network approach for assisting avalanche search and rescue operations with UAV imagery. *Remote Sensing*, 9(2). <https://doi.org/10.3390/rs9020100>
- Cholissodin, I., & Soebroto, A. A. (2020). *AI, MACHINE LEARNING & DEEP LEARNING (Teori & Implementasi)*. July 2019.
- Hu, F., Xia, G. S., Hu, J., & Zhang, L. (2015). Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery. *Remote Sensing*, 7(11), 14680–14707. <https://doi.org/10.3390/rs71114680>
- Pangestu, M. A., & Bunyamin, H. (2018). Analisis Performa dan Pengembangan Sistem Deteksi Ras Anjing pada Gambar dengan Menggunakan Pre-Trained CNN Model. *Jurnal Teknik Informatika Dan Sistem Informasi*, 4, 337–344.
- Setiawan, W. (2020). Perbandingan Arsitektur Convolutional Neural Network Untuk Klasifikasi Fundus. *Jurnal Simantec*, 7(2), 48–53. <https://doi.org/10.21107/simantec.v7i2.6551>
- Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 1–14.
- Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., & Liu, C. (2018). A survey on deep transfer learning. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 11141 LNCS, 270–279. https://doi.org/10.1007/978-3-030-01424-7_27