

Klasifikasi Gambar dari Prototipe Camera Trap Menggunakan Model ResNet-50 untuk Mendeteksi Satwa Dilindungi

Arif Bastanta Sinuhaji, Aji Gautama Putrada², Hilal Hudan Nuha³,

^{1,2,3} Universitas Telkom, Bandung

¹arifbastanta@student.telkomuniversity.ac.id, ²ajigps@telkomuniversity.ac.id,

³hilalnuha@telkomuniversity.ac.id

Abstrak

Indonesia merupakan salah satu negara dengan keanekaragaman flora dan fauna yang tersebar diberbagai wilayah. Dalam keanekaragaman tersebut terdapat beberapa satwa yang memerlukan perhatian khusus karena statusnya yang mulai langka. Pendekatan teknologi menggunakan *camera trap* merupakan salah satu upaya yang telah dilakukan untuk memonitor satwa-satwa langka tersebut. Tugas akhir ini bertujuan untuk melakukan klasifikasi gambar yang ditangkap oleh sebuah prototipe *camera trap* dengan menggunakan salah satu model arsitektur *Convolutional Neural Network* (CNN) yaitu *Residual Network-50* (ResNet-50). *Convolutional Neural network* merupakan salah satu metode yang banyak digunakan pada metode *deep learning* yang cara kerjanya meniru jaringan syaraf otak manusia. Pada tugas akhir ini dilakukan eksperimen berupa beberapa dataset dengan fitur yang berbeda dalam setiap skenarionya. Eksperimen menghasilkan sebuah model yang paling optimal dengan akurasi *training* sebesar 99,34 % dan akurasi testing sebesar 90,43 %.

Kata kunci : *deep learning, camera trap, ResNet-50, CNN, satwa.*

Abstract

Indonesia is one of the countries with a diversity of flora and fauna spread in various regions. In this diversity there are several animals that require special attention because of their rare status. The technological approach using camera traps is one of the efforts that have been made to monitor these endangered animals. This final project aims to classify images captured by a camera trap prototype using one of the *Convolutional Neural Network* (CNN) architectural models, namely *Residual Network-50* (ResNet-50). *Convolutional Neural Network* is one of the methods that is widely used in deep learning methods which works by imitating the neural network of the human brain. In this final project, experiments are carried out in the form of several datasets with different features in each scenario. The experiment resulted in the most optimal model with training accuracy of 99.34% and testing accuracy of 90.43%.

Keywords: *deep learning, camera trap, ResNet-50, CNN, animals.*

1. Pendahuluan

Latar Belakang

Keanekaragaman flora dan fauna tersebar di berbagai wilayah yang terletak di seluruh permukaan bumi ini. Indonesia merupakan salah satu negara yang wilayahnya dipenuhi dengan berbagai macam spesies satwa di dalamnya, termasuk beberapa satwa yang memerlukan perhatian khusus dikarenakan populasinya tinggal sedikit sehingga statusnya dilindungi [2]. Menurut Peraturan Menteri Lingkungan Hidup Dan Kehutanan Republik Indonesia Nomor P.106/MENLHK/SETJEN/KUM.1/12/2018 disebutkan beberapa spesies satwa tersebut antara lain harimau, badak, orangutan, gajah, beruang, dll. Berbagai upaya dilakukan oleh pemerintah maupun kalangan lainnya untuk melestarikan satwa-satwa dengan status dilindungi atau terancam punah tersebut agar populasinya tetap terjaga.

Salah satu upaya yang dilakukan yaitu dengan menggunakan pendekatan teknologi, pada tugas akhir ini dilakukan klasifikasi pada beberapa kelas binatang menggunakan metode *deep learning* yaitu *Convolutional Neural Network* atau jaringan syaraf tiruan yang cara kerjanya menyerupai otak manusia. Pada tugas akhir ini dibuat sebuah prototipe *camera trap* yang berfungsi untuk mengambil gambar dari satwa-satwa tersebut. Saat ini, *camera trap* merupakan salah satu alat yang banyak digunakan untuk memonitor satwa-satwa yang berada di alam liar [12]. *Camera Trap* merupakan metode standar untuk mengamati pergerakan dari berbagai spesies satwa di hutan [20]. Penelitian menggunakan *camera trap* memerlukan pengamatan gambar secara visual untuk mendapatkan data yang dapat digunakan dalam menganalisis ekologis [21]. Dalam *Deep Learning*, terdapat berbagai varian model arsitektur yang menggunakan CNN misalnya VGG16, InceptionV3, Xception, ResNet50, dll. Pada tugas akhir ini digunakan salah satu model yang dikenal baik untuk pengolahan gambar yaitu menggunakan model arsitektur ResNet50.

Disebutkan oleh seorang penulis bernama Sharma bahwa ResNet50 mempunyai performansi rata-rata yang lebih optimal dibandingkan dengan AlexNet dan GoogLeNet yaitu sebesar 78.10 % [3]. Pada tugas akhir ini dilakukan beberapa eksperimen berupa dataset dengan fitur yang berbeda untuk setiap skenarionya. Eksperimen tersebut dilakukan untuk mengetahui model dengan skenario fitur mana yang paling optimal. Model yang paling optimal akan digunakan untuk mengklasifikasi gambar dari *trap camera* dengan tujuan mendeteksi satwa yang dilindungi.

Topik dan Batasannya

Implementasi pada model yang dibangun digunakan untuk mengklasifikasi gambar dari prototipe *camera trap* sehingga tugas akhir ini difokuskan untuk dapat mendeteksi kelas satwa dilindungi yang tertangkap oleh kamera. Batasan masalah pada tugas akhir ini adalah sebagai berikut:

1. Modul mikrokontroler yang digunakan pada prototipe *camera trap* adalah ESP32-Cam yang membutuhkan koneksi internet.
2. Dataset yang digunakan sekitar 125.000 data gambar dengan format JPG resolusi 256x256.
3. *Pretrained* model yang digunakan adalah ResNet-50
4. Gambar dari kelas satwa yang dilindungi berisikan gambar satwa dengan spesies badak, beruang, gajah, harimau dan orangutan.
5. Gambar dari kelas objek yang tidak dideteksi berisikan gambar manusia, babi hutan, rusa, anjing, kuda, bajing dan kelinci.
6. Skenario fitur pada gambar dataset yang digunakan memiliki lima jenis yaitu :
 - (a) Gambar dengan perspektif penuh pada fitur yang dimiliki satwa.
 - (b) Gambar dengan perspektif sebagian pada fitur yang dimiliki satwa.
 - (c) Gambar dengan perspektif sangat dekat pada satwa.
 - (d) Gambar dengan perspektif penuh pada fitur yang dimiliki satwa namun dengan pencahayaan redup.
 - (e) Gambar dengan perspektif penuh pada fitur yang dimiliki satwa namun pewarnaan *grayscale*.
7. Pengujian dilakukan dengan mengarahkan *camera trap* pada layar TV yang menjalankan video dari kelas satwa yang dilindungi.

Tujuan

Tujuan dari tugas akhir ini melakukan klasifikasi gambar dari prototipe *camera trap* untuk mendeteksi kelas satwa yang dilindungi dengan menggunakan *pretrained* model ResNet-50 yang kemudian akan dilakukan evaluasi untuk mendapatkan model paling optimal berdasarkan dataset dengan skenario fitur terbaik yang dimiliki oleh gambar.

Organisasi Tulisan

Terdapat 5 bagian pada tugas akhir ini. Bagian pertama mengenai latar belakang mengapa tugas akhir ini menggunakan prototipe *camera trap* dan *deep learning* untuk mendeteksi satwa dilindungi. Pada bagian kedua berisikan penjelasan mengenai studi terkait dengan tugas akhir. Pada bagian ketiga dijelaskan mengenai perancangan prototipe *camera trap* dan pembangunan model. Pada bagian keempat dijelaskan mengenai hasil dan evaluasi dari sistem yang dibangun pada tugas akhir. Pada bagian kelima merupakan kesimpulan dari tugas akhir ini dan saran untuk penelitian selanjutnya.

2. Studi Terkait

Camera Trap

Camera Trap semakin banyak digunakan untuk mengamati dan mempelajari satwa liar di hutan [12]. Gray melakukan pemantauan terhadap ungulate hutan tropis menggunakan *Camera Trap* [19]. *Camera trap* dilengkapi dengan baterai dan memori untuk menyimpan data dari gambar ataupun video yang ditangkap [13]. Data yang diambil oleh *camera trap* juga dapat digunakan untuk berbagai macam penelitian, misalnya digunakan untuk proses pembangunan model arsitektur CNN [1]. Model tersebut nantinya dapat digunakan untuk melakukan pendeteksian satwa secara otomatis [14] seperti prototipe yang dibuat pada tugas akhir ini.

Convolutional Neural Network

Convolutional Neural Network (CNN) merupakan *deep neural network* yang biasa digunakan untuk pengenalan gambar [8]. Cara kerja *Convolutional Neural Network* meniru dari jaringan syaraf otak manusia [7].

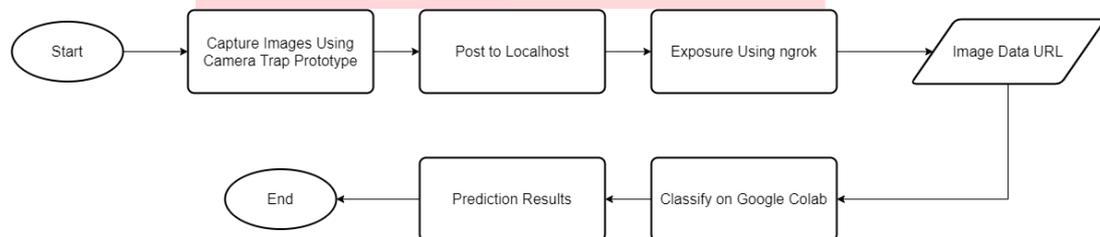
Miao Z (2019) melakukan penelitian untuk konservasi dan pengelolaan satwa menggunakan menggunakan CNN dan mendapatkan akurasi sebesar 87,5 % [5]. Seorang penulis Bernama Duhart C. (2019) juga melakukan penelitian dengan skala besar, menghasilkan akurasi sebesar 96,8 % menggunakan VGG dengan model yang dilatih dengan 1.4 juta gambar. Penelitian khusus untuk mengenali burung coklat juga pernah dilakukan dan mendapatkan hasil berupa presisi rata-rata 98 % dan akurasi sebesar 83,9% [9]. Pernah dilakukan pengenalan dan identifikasi untuk *monitoring* margasatwa secara otomatis dan menunjukkan akurasi yang tinggi sebesar 96 % untuk pendeteksian satwa [11]. Chen R. (2019) pernah melakukan multiklasifikasi dan mendapatkan hasil sebesar 90,32 %.

ResNet-50

Residual Network-50 (ResNet-50) merupakan salah satu model arsitektur CNN yang sangat baik sehingga banyak digunakan [15]. Reddy A. (2019) pernah melakukan penelitian menggunakan ResNet-50 mengklasifikasi data sel penyakit malaria menggunakan metode *transfer learning* [16]. Rajpal S. (2021) juga melakukan identifikasi terkait COVID-19 melalui rontgen dari bagian dada manusia dan menghasilkan akurasi sebesar 97,42 % [17]. Varian model ResNet mampu memprediksi kanker kolorektal dengan akurasi antara 73 % - 88 % dan sensitivitas 64 % - 96 % [23].

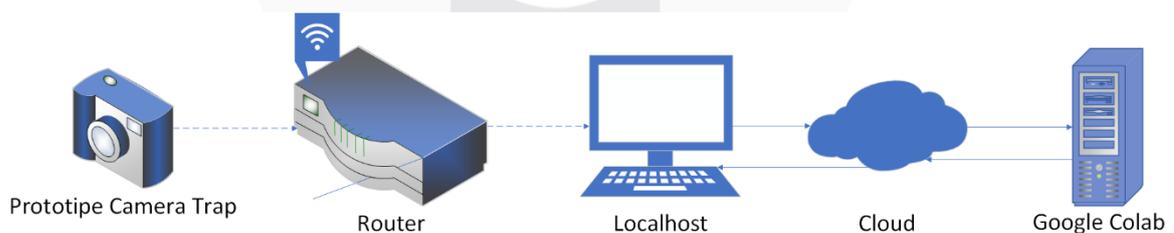
3. Sistem yang Dibangun

3.1 Gambaran Umum Sistem



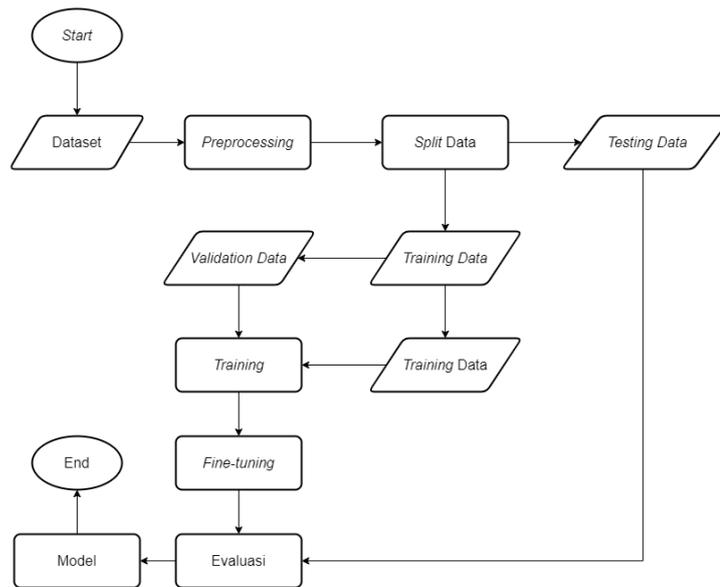
Gambar 1 Gambaran umum sistem

Gambar diatas merupakan desain *flowchart* dari sistem yang dibangun. Mikrokontroler yang berupa prototipe dari *camera trap* akan melakukan pengambilan gambar. Gambar di dikirimkan pada server lokal atau *localhost* yang kemudian diekspos menggunakan ngrok menjadi sebuah URL yang dapat diakses dengan internet. Jika hasil klasifikasi dengan model tersebut tidak terdeteksi sebagai satwa yang dilindungi maka gambar tidak akan ditampilkan. Sebaliknya, jika gambar tersebut terdeteksi sebagai satwa dilindungi maka gambar akan ditampilkan. Topologi sistem yang dibangun dapat dilihat pada gambar di bawah ini.



Gambar 2 Topologi Sistem

3.1.1 Alur Pembangunan Model

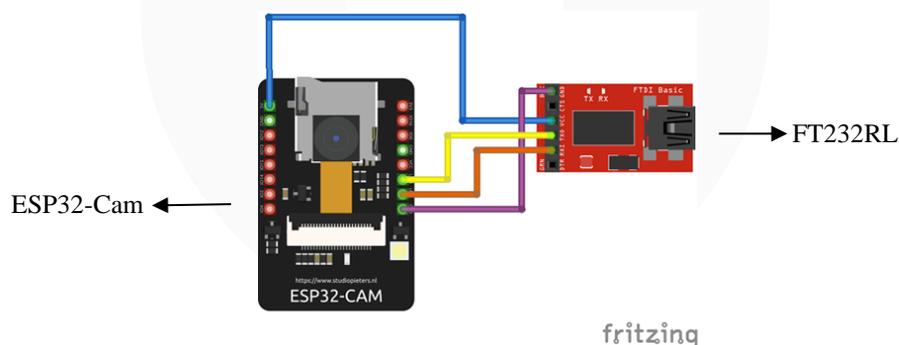


Gambar 3 Flowchart Pembangunan Model

Proses pembangunan model dilakukan pada *tools* berbasis *cloud* yaitu *Google Colab*. Pada proses membangun model diawali dengan melakukan pengumpulan dataset berupa gambar yang kemudian dilakukan preprocessing dengan melakukan augmentasi pada data. Selanjutnya data akan di split menjadi data training dan data testing dengan rasio perbandingan 80:20. Kemudian data training yang telah di split akan dilatih. Setelah proses training selesai maka akan dilanjutkan kepada tahapan selanjutnya yaitu dengan melakukan Fine-tuning. Setelah itu akan dilakukan evaluasi dengan menguji hasil data yang telah dilatih dengan data testing. Selanjutnya akan terbuat model yang telah dilakukan penambahan kelas berupa lima kelas satwa yang dilindungi dan satu kelas berisikan kumpulan objek bukan satwa dilindungi.

3.2 Mikrokontroler

3.2.1 Perancangan Mikrokontroler



Gambar 4 Desain Mikrokontroler

Desain dari perancangan prototipe *Camera Trap* di atas dibuat menggunakan aplikasi fritzing. Prototipe dari *Camera Trap* dirancang menggunakan dua buah modul mikrokontroler yaitu modul ESP32-Cam dan FT232RL yang dihubungkan menggunakan kabel *jumper*. Modul FT232RL mempunyai 6 buah pin yang mana pin tersebut digunakan untuk menghubungkan modul FT232RL ke modul ESP32-Cam. Namun hanya terdapat 4 buah pin yang dihubungkan yaitu pin RX pada modul FT232RL dihubungkan ke pin U0T pada modul ESP32-Cam, pin TX pada modul FT232RL dihubungkan ke pin U0R pada modul ESP32-Cam, pin VCC pada modul FT232RL dihubungkan ke pin 5V pada modul ESP32-Cam dan terakhir pin GND pada modul FT232RL dihubungkan ke pin GND pada modul ESP32-Cam.

3.2.2 Perolehan Data Gambar

Prototipe *camera trap* akan melakukan proses pengambilan gambar dalam kurun waktu setiap 1000 ms atau 1 detik dan mengirimkannya kepada server lokal. Gambar yang telah ditangkap dapat diperoleh dengan menuliskan 'ipaddress/image' pada *browser*. Kemudian agar data gambar tersebut dapat diakses melalui internet maka akan digunakan ngrok untuk mengekspos server lokal ke internet publik. Setelahnya akan didapatkan sebuah URL berisikan data gambar yang kemudian akan dilakukan klasifikasi untuk mendeteksi gambar tersebut termasuk gambar dari satwa yang dilindungi atau tidak.

3.2.3 Pengujian Mikrokontroler

Pengujian mikrokontroler berupa prototipe *camera trap* dilakukan dengan mengarahkan modul kamera ESP32-Cam kepada sebuah layar monitor yang menampilkan video dari satwa yang bergerak. Prototipe *camera trap* ini akan diuji seberapa baik dalam melakukan pengambilan gambar dari video yang bergerak. Prototipe *camera trap* diarahkan pada sebuah layar TV yang memutar video bergerak dari kelas-kelas satwa dilindungi dan objek yang tidak teridentifikasi satwa dilindungi.

3.3 Model yang Dibangun

3.3.1 Pengumpulan Dataset Gambar

Dataset gambar diperoleh dari *AwA2*, *Kaggle* dan video dari *youtube*. Dataset yang digunakan terdiri dari lima jenis spesies satwa yang dilindungi di negara Indonesia yaitu badak, beruang, gajah, harimau dan orangutan. Dataset ini nantinya digunakan untuk melatih serta menguji model yang akan mengklasifikasikan satwa-satwa dilindungi yang telah disebutkan tadi. Dataset ini juga berisi objek-objek yang tidak diidentifikasi sebagai satwa yang dilindungi seperti manusia, babi hutan, rusa, anjing, kuda, bajing dan kelinci. Dataset ini akan dilabelkan dengan menempatkan gambar pada direktori dan memberi nama pada gambar sesuai kelas dan fungsionalitasnya. Salah satu kasus yaitu pada gambar gajah maka akan dilabelkan dengan nama gajah dan nomornya seperti contoh : GajahTrain1.jpg, GajahTrain2.jpg, GajahTrain3.jpg dan seterusnya. Pada gambar yang tidak diidentifikasi sebagai satwa yang dilindungi maka akan dilabelkan dengan nama NonDeteksiTrain.jpg.

Terdapat dataset yang telah disiapkan untuk menguji model yang telah dilatih sesuai dengan perspektif fitur yang berbeda-beda dan dapat dilihat pada tabel di bawah ini :

Tabel 1 Skenario pengujian

Skenario	Dataset	Keterangan
1	A	Gambar menampilkan seluruh bagian tubuh satwa dengan lengkap
2	B	Gambar menampilkan sebagian tubuh satwa
3	C	Gambar menampilkan satwa dengan sangat dekat
4	D	Gambar menampilkan seluruh bagian tubuh satwa dengan lengkap tetapi dengan pencahayaan yang redup
5	E	Gambar menampilkan seluruh bagian tubuh satwa dengan lengkap tetapi dengan pewarnaan <i>grayscale</i>



(A)



(B)



(C)



(D)



(E)

3.3.2 Splitting Data

Terdapat dataset sekitar 125.000 gambar yang memiliki perspektif yang berbeda berdasarkan fiturnya. Seperti yang sudah disebutkan sebelumnya, dilakukan eksperimen untuk menemukan model yang paling optimal berdasarkan lima skenario fitur. Data gambar sekitar 125.000 akan dibagi lima sesuai fitur masing-masing yang dimiliki gambar. Setiap model akan memiliki data sekitar sebanyak 25.000 gambar yang kemudian akan dibagi lagi menjadi data *training* dan data *testing* dengan rasio perbandingan sebesar 80:20. Kemudian data *training* akan

dibagi lagi menjadi data *training* dan data *validation* dengan rasio perbandingan 90:10. Hal ini berguna untuk mengecek akurasi pada data *training* dan akurasi pada data *validation*. Data *training* digunakan untuk melatih model sedangkan data *validation* digunakan untuk mencegah terjadinya *overfitting*. Pada tahap ini disediakan data *testing* gabungan dengan mengambil 1/5 gambar dari setiap data *testing* sesuai dengan skenario yang telah dijelaskan pada Tabel 1.

3.3.3 Preprocessing Data

Sebelum data masuk pada proses *training*, data mengalami proses augmentasi terlebih dahulu agar model dilatih dan dapat memprediksi perspektif dari gambar satwa yang berbeda-beda. Dataset gambar dilakukan augmentasi dengan menggunakan suatu kelas dari *library keras* yaitu *ImageDataGenerator*. Proses augmentasi data dilakukan pada gambar yang dipilih secara acak dengan parameter sebagai berikut :

- Mengubah setiap nilai piksel dari *range* [0,255]
- Menggeser gambar searah sumbu X sebesar 20%
- Menggeser gambar searah sumbu Y sebesar 20%
- Memutar gambar sebesar 30 derajat
- Memperbesar gambar sebesar 20%
- *Shear* gambar skala 20%
- Memutar gambar secara horizontal
- Mengisi ruang piksel yang kosong dengan piksel terdekat.

3.3.4 Training Data dengan Model ResNet-50

Setelah data melewati tahap *preprocessing* dan *splitting* data, data *training* akan masuk ke dalam *pretrained* model ResNet-50 untuk dilatih. Saat pelatihan model, data *training* akan dibagi menjadi data *training* dan data *validation* seperti yang sudah dijelaskan sebelumnya. Pengantisipasi terjadinya *overfitting* dilakukan dengan memantau peningkatan akurasi *training* dan akurasi *validation*. Pada konfigurasi model ResNet-50 dilakukan *hyperparameter tuning* untuk mengatur jalannya *training* model tersebut. *Hyperparameter* yang akan dikonfigurasi untuk menjalankan *learning* adalah *epoch*, *loss function*, *optimizer* dan *batch*. Setelah itu dilakukan *load* model ResNet-50, model akan ditambahkan beberapa layer tambahan beserta jumlah neuron dan fungsi aktivasinya untuk mengklasifikasi data yang akan dilatih.

3.3.5 Pengukuran Kinerja Model

Untuk melakukan pengukuran kinerja dari model yang dibangun menggunakan *confusion matrix*. Pengukuran kinerja yang model pada *confusion matrix* berupa akurasi yang menggunakan perhitungan dari jumlah TP, TN, FP dan FN. Perhitungan tersebut dilakukan dengan menggunakan persamaan 1 [22].

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN} \quad (1)$$

Model yang telah dilatih akan dilakukan uji coba menggunakan data *testing* yang merupakan dataset gabungan. Dataset gabungan merupakan data yang terdiri dari setiap gambar dengan fitur yang berbeda-beda seperti yang sudah dijelaskan pada Tabel 1.

4. Evaluasi

4.1 Hasil Pengujian

Model yang telah dilatih dengan dataset gambar sesuai dengan fitur yang dimilikinya maka akan didapatkan hasil berupa akurasi *training* dan akurasi *validation*. Model dilatih selama 10 epoch dengan *batch* sebanyak 32. Proses dilakukan pada *Google Colab* dengan spesifikasi CPU Intel® Xeon® CPU @2.30GHz, RAM 13 GB, GPU Tesla K80 12 GB dan Sistem Operasi 64-bit. Pada Tabel 2 dapat dilihat tingkat akurasi *training* dan akurasi *validation* paling baik terdapat pada skenario 1 dengan akurasi *training* sebesar 99,34 % dan akurasi *validation* sebesar 99,71 % yang dapat dilihat pada tabel dibawah ini :

Tabel 2 Hasil *training*

Skenario	Akurasi <i>Training</i>	Akurasi <i>Validation</i>
Skenario 1	99,34 %	99,71 %
Skenario 2	97,89 %	98,87 %
Skenario 3	93,34 %	95,41 %
Skenario 4	99,15 %	99,61 %
Skenario 5	99,08 %	99,51 %

Untuk hasil pengukuran kinerja model menggunakan akurasi dengan data *testing* gabungan menggunakan perbandingan yang sudah dijelaskan sebelumnya. Didapatkan model hasil dari proses pengujian dilihat pada Tabel 3 di bawah ini :

Tabel 3 Pengukuran kinerja model

Skenario	Akurasi <i>Testing</i>
Skenario 1	90,43 %
Skenario 2	85,71 %
Skenario 3	55,36 %
Skenario 4	33,80 %
Skenario 5	50,93 %

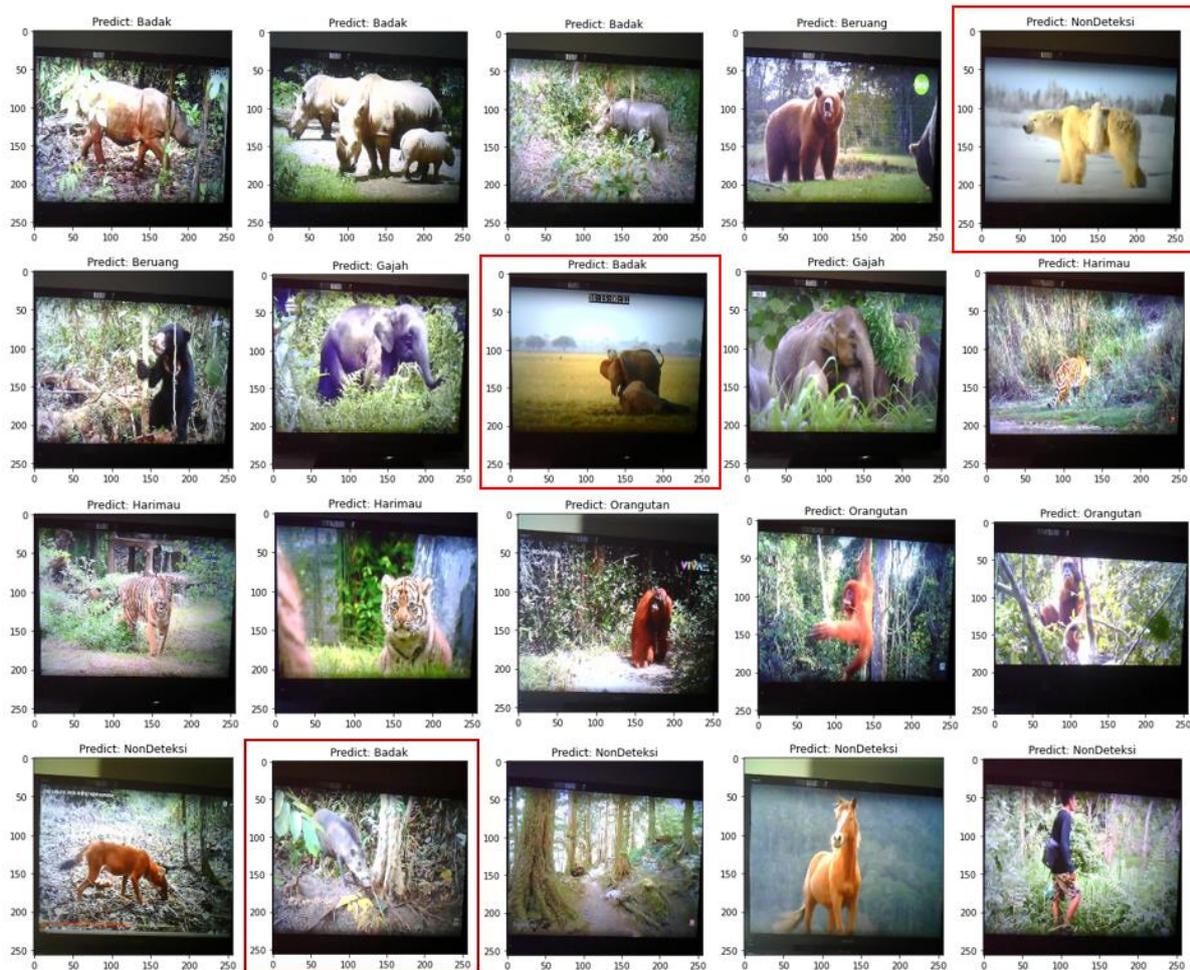
Berdasarkan data hasil pengujian kinerja model pada Tabel 3, model yang paling optimal terdapat pada model dengan skenario 1. Model dengan skenario 1 yang memiliki akurasi sebesar 90,43%, sedangkan model dengan hasil akurasi terburuk terdapat pada model dengan skenario 5 yang memiliki akurasi sebesar 50,93 %.

Pada Tabel 5 ditunjukkan *True Positive Rate* (TPR), *False Negative Rate* (FNR) dan *False Positive Rate* (FPR) untuk setiap kelasnya [1] yaitu sebagai berikut :

Tabel 4 Hasil TPR, FNR dan FPR

Kelas	TPR	FNR	FPR
Badak	83,01 %	16,99 %	0,38 %
Beruang	93,61 %	6,39 %	3,58 %
Gajah	90,81 %	9,19 %	3,89 %
Harimau	95,71 %	4,29 %	0,79 %
NonDeteksi	91,25 %	8,75 %	2,40 %
Orangutan	88,52 %	11,48 %	0,44 %

Selanjutnya pengujian terhadap prototipe *camera trap* dalam proses pengambilan gambar dari video berupa satwa yang bergerak. Gambar tersebut dideteksi menggunakan model yang paling optimal yaitu model dengan skenario 1 dan mendapatkan hasil pada Gambar 4 sebagai berikut.



Gambar 5 Hasil prediksi dari proses klasifikasi satwa dilindungi. Gambar yang diberi kotak merah adalah gambar dari kelas satwa yang tidak terprediksi sesuai kelasnya.

4.2 Analisis Hasil Pengujian

4.2.1 Analisis Hasil Pengujian Proses *Training*

Menurut data yang ditampilkan pada Tabel 2 berupa akurasi dari proses *training* pada setiap skenario menunjukkan hasil yang paling baik. Setiap model dengan skenarionya tersendiri memiliki hasil akurasi *training* dan akurasi *validasi* yang tinggi. Hal ini terjadi karena dataset yang digunakan pada setiap skenario menggunakan data gambar dengan perspektif sesuai fitur yang dimiliki oleh gambar-gambar tersebut. Namun, dapat dilihat model yang paling optimal dalam proses *training* adalah model dengan skenario 1 yang memiliki akurasi *train* sebesar 99,34 % dan akurasi *validation* 99,71 % karena menggunakan dataset berisikan gambar-gambar dari satwa yang memiliki fitur lengkap. Jika dibandingkan dengan model skenario 3 yang memiliki akurasi *train* sebesar 93,34 % dan akurasi *validation* 95,41 %, model dengan skenario 3 memiliki nilai akurasi yang paling kecil dibandingkan model dengan skenario lainnya. Hal ini disebabkan karena model dengan skenario 3 melakukan proses *training* menggunakan dataset dengan skenario gambar satwa dari dekat yang memiliki ketidakjelasan fitur pada gambar untuk dipelajari oleh model.

4.2.2 Analisis Hasil Pengukuran Kinerja Model

Pada Tabel 3 ditampilkan data berupa skenario beserta akurasi yang dihasilkan berdasarkan data *testing* gabungan. Model yang dengan skenario 1 memiliki akurasi *testing* terbaik yang menghasilkan akurasi *testing* sebesar 90,43 %. Hal ini membuktikan bahwa model dengan skenario 1 merupakan model yang paling optimal karena dapat memprediksi lebih banyak gambar dibandingkan model dengan skenario lainnya. Hal ini bisa terjadi karena model dengan skenario 1 dilatih dengan dataset yang berisikan gambar dari satwa dengan fitur lengkap

seperti wajah, badan, kaki, ekor, warna, dll. Namun, berdasarkan pengujian tersebut dapat diketahui bahwa fitur berupa warna memiliki peranan yang penting. Hal ini dibuktikan pada model dengan skenario 4 yang memiliki akurasi terkecil sebesar 33.80 %. Hal ini terjadi karena model dengan skenario 4 dilatih menggunakan dataset yang memiliki pencahayaan redup. Menariknya, pada model dengan skenario 5 yang bahkan memiliki dataset berisikan gambar dengan pewarnaan *grayscale*, menghasilkan akurasi yang lebih baik dari model dengan skenario 3 yaitu 50.93 %. Hal ini dapat terjadi karena terdapat kelas dari spesies satwa badak dan gajah yang memiliki dominan warna keabuan pada dua satwa tersebut.

4.2.3 Analisis Hasil Prediksi

Berdasarkan hasil prediksi oleh model yang ditampilkan pada Gambar 4, model dapat melakukan klasifikasi dengan cukup baik. Dari 20 gambar, 17 gambar diantaranya diprediksi sesuai kelasnya dengan benar. Namun, terdapat 3 gambar yang diberi kotak warna merah dengan hasil prediksinya salah. Pertama, gambar beruang kutub diprediksi NonDeteksi yang seharusnya diprediksi Beruang oleh model. Hal ini dapat terjadi karena model tidak dilatih dengan gambar beruang kutub yang memiliki intensitas warna didominasi dengan warna putih. Kedua, gambar gajah diprediksi Badak yang seharusnya diprediksi Gajah oleh model. Hal tersebut dapat terjadi karena gajah dan badak memiliki kemiripan yaitu pada warna kulitnya yang keabuan serta gajah yang memiliki gading dan badak yang memiliki cula. Faktor dari sudut pandang pengambilan gambar juga mempengaruhi hasil prediksi oleh model. Ketiga, gambar babi hutan diprediksi Badak yang seharusnya diprediksi NonDeteksi. Sama halnya dengan kasus kedua sebelumnya, jika dilihat pada Gambar 4 sekilas babi hutan tersebut memiliki kemiripan dengan badak yaitu warnanya yang keabuan dan bentuk dari tubuhnya. Sehingga tingkat keberhasilan yang dimiliki model dengan memprediksi 20 gambar tersebut yaitu sebesar 85 %. Tabel dibawah ini merupakan waktu pemrosesan yang di bandingkan dengan model ResNet-50 :

Tabel 4 Execution Time

<i>Methods</i>	<i>Time (ms)</i>
Mask R-CNN [24]	254.35
YOLO [25]	20.42
Proposed [18]	24.01
ResNet-50	1.900

5. Kesimpulan

Berdasarkan hasil eksperimen yang dilakukan pada tugas akhir ini, dapat disimpulkan model ResNet-50 yang dilatih menggunakan dataset dengan perspektif gambar binatang dengan fitur lengkap memiliki akurasi yang paling tinggi yaitu sebesar 99,34 % untuk akurasi *training* dan 90,43 % untuk akurasi *testing*. Prototipe *camera trap* yang digunakan dapat melakukan pengambilan gambar dari video dari satwa yang bergerak.

Saran untuk penelitian selanjutnya dapat menggunakan modul mikrokontroler kamera yang memiliki kualitas gambar lebih baik dan melakukan penambahan untuk kelas-kelas satwanya. Untuk pengaplikasian lebih lanjut, sistem yang dibangun mungkin dapat dilakukan otomatisasi dengan menggunakan internet satelit untuk akses internet di hutan.

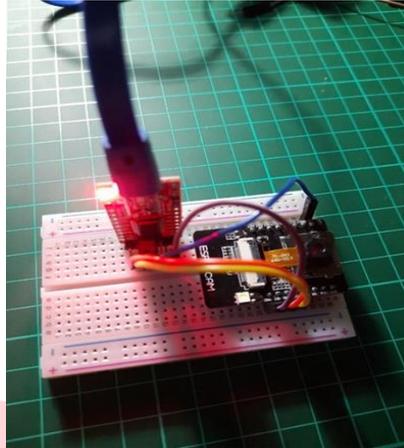
Referensi

- [1] Favorskaya, M., & Pakhirka, A. (2019). Animal species recognition in the wildlife based on muzzle and shape features using joint CNN. *Procedia Computer Science*, 159, 933-942.
- [2] Jadda, A. A. (2019). Tinjauan Hukum Lingkungan Terhadap Perlindungan dan Pengelolaan Keanekaragaman Hayati. *Madani Legal Review*, 3(1), 39-62.
- [3] Sharma, N., Jain, V., & Mishra, A. (2018). An analysis of convolutional neural networks for image classification. *Procedia computer science*, 132, 377-384.
- [4] Çinar, A., & Yildirim, M. (2020). Detection of tumors on brain MRI images using the hybrid convolutional neural network architecture. *Medical hypotheses*, 139, 109684.
- [5] Miao, Z., Gaynor, K. M., Wang, J., Liu, Z., Muellerklein, O., Norouzzadeh, M. S., ... & Getz, W. M. (2019). Insights and approaches using deep learning to classify wildlife. *Scientific reports*, 9(1), 1-9.
- [6] Duhart, C., Dublon, G., Mayton, B., Davenport, G., & Paradiso, J. A. (2019, June). Deep learning for wildlife conservation and restoration efforts. In 36th International Conference on Machine Learning, Long Beach (Vol. 5).
- [7] Mehta, R., Majumdar, A., & Sivaswamy, J. (2017). BrainSegNet: a convolutional neural network architecture for automated segmentation of human brain structures. *Journal of Medical Imaging*, 4(2), 024003.
- [8] Kim, P. (2017). Convolutional neural network. In *MATLAB deep learning* (pp. 121-147). Apress, Berkeley, CA.
- [9] Clapham, M., Miller, E., Nguyen, M., & Darimont, C. T. (2020). Automated facial recognition for wildlife that lack unique markings: A deep learning approach for brown bears. *Ecology and evolution*, 10(23), 12883-12892.
- [10] Nguyen, H., Maclagan, S. J., Nguyen, T. D., Nguyen, T., Flemons, P., Andrews, K., ... & Phung, D. (2017, October). Animal recognition and identification with deep convolutional neural networks for automated wildlife monitoring. In 2017 IEEE international conference on data science and advanced Analytics (DSAA) (pp. 40-49). IEEE.
- [11] Chen, R., Little, R., Mihaylova, L., Delahay, R., & Cox, R. (2019). Wildlife surveillance using deep learning methods. *Ecology and evolution*, 9(17), 9453-9466.
- [12] Sollmann, R. (2018). A gentle introduction to camera-trap data analysis. *African Journal of Ecology*, 56(4), 740-749.
- [13] Janzen, M., Visser, K., Visscher, D., MacLeod, I., Vujnovic, D., & Vujnovic, K. (2017). Semi-automated camera trap image processing for the detection of ungulate fence crossing events. *Environmental monitoring and assessment*, 189(10), 1-13.
- [14] Schneider, S., Taylor, G. W., & Kremer, S. (2018, May). Deep learning object detection methods for ecological camera trap data. In 2018 15th Conference on computer and robot vision (CRV) (pp. 321-328). IEEE.
- [15] Al-Haija, Q. A., & Adebajo, A. (2020, September). Breast cancer diagnosis in histopathological images using ResNet-50 convolutional neural network. In 2020 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS) (pp. 1-7). IEEE.
- [16] Reddy, A. S. B., & Juliet, D. S. (2019, April). Transfer learning with ResNet-50 for malaria cell-image classification. In 2019 International Conference on Communication and Signal Processing (ICCSP) (pp. 0945-0949). IEEE.
- [17] Rajpal, S., Lakhyani, N., Singh, A. K., Kohli, R., & Kumar, N. (2021). Using handpicked features in conjunction with ResNet-50 for improved detection of COVID-19 from chest X-ray images. *Chaos, Solitons & Fractals*, 145, 110749.
- [18] Lee, H., Sa, J., Chung, Y., Park, D., & Kim, H. (2019). Deep Learning-based Overlapping-Pigs Separation by Balancing Accuracy and Execution Time.
- [19] Gray, T. N. E. (2018). Monitoring tropical forest ungulates using camera-trap data. *Journal of Zoology*, 305(3), 173-179.
- [20] Kays, R., Arbogast, B. S., Baker-Whatton, M., Beirne, C., Boone, H. M., Bowler, M., ... & Spironello, W. R. (2020). An empirical evaluation of camera trap study design: How many, how long and when?. *Methods in Ecology and Evolution*, 11(6), 700-713.
- [21] Tabak, M. A., Norouzzadeh, M. S., Wolfson, D. W., Sweeney, S. J., VerCauteren, K. C., Snow, N. P., ... & Miller, R. S. (2019). Machine learning to classify animal species in camera trap images: Applications in ecology. *Methods in Ecology and Evolution*, 10(4), 585-590.
- [22] Chicco, D., Tötsch, N., & Jurman, G. (2021). The Matthews correlation coefficient (MCC) is more reliable than balanced accuracy, bookmaker informedness, and markedness in two-class confusion matrix evaluation. *BioData mining*, 14(1), 1-22.

- [23] Sarwinda, D., Paradisa, R. H., Bustamam, A., & Anggia, P. (2021). Deep Learning in Image Classification using Residual Network (ResNet) Variants for Detection of Colorectal Cancer. *Procedia Computer Science*, 179, 423-431.
- [24] He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision* (pp. 2961-2969).
- [25] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 779-788).



Lampiran



Gambar 6 Prototipe *Camera Trap*



Gambar 7 Proses Uji Pendeteksian