

# Klasifikasi Gambar Satwa Liar Dengan Menggunakan Metode *Convolutional Neural Networks* (CNN) Studi Kasus Taman Nasional Bukit Barisan Selatan

Ira Safitri

Magister Sains Komputasi Institut Teknologi Bandung, Jl. Ganesa No.10, Kota Bandung,  
Jawa Barat, Indonesia

Email: [20920004@mahasiswa.itb.ac.id](mailto:20920004@mahasiswa.itb.ac.id)

**Abstrak.** Taman Nasional Bukit Barisan Selatan (TNBBS) merupakan salah satu hutan konservasi warisan dunia yang memiliki ekosistem hutan hujan tropis yang lengkap. TNBBS memiliki kegiatan konservasi ekologi yang bekerjasama dengan *Wildlife Conservation Society Indonesia Program* (WCS-IP) yang bertujuan dalam pemantauan satwa liar. Penelitian sebelumnya telah menggunakan kamera pengawas dalam pemantauan satwa liar secara otomatis menggunakan metode *Convolutional Neural Networks*. Hal ini menyebabkan diperlukan penelitian lanjut terkait klasifikasi satwa liar berdasarkan data gambar menggunakan metode *Convolutional Neural Networks* (CNN). Penelitian ini akan dilakukan di lokasi TNBBS dengan memasang kamera pengawas sehingga data gambar dapat diklasifikasi menggunakan metode *Convolutional Neural Networks* (CNN). Dengan penelitian ini diharapkan dapat diketahui tingkat akurasi model yang diperoleh dan seberapa mampu model CNN mengklasifikasi gambar satwa liar dengan akurat. Analisis yang dilakukan diperoleh perbandingan akurasi dari 2 jenis optimizer yaitu 95,22% untuk optimizer Adam pada epoch sebesar 50 sedangkan pada epoch 100 didapatkan akurasi 96,74%. Pada optimizer RMSProp dengan epoch 50 epoch dihasilkan 96,03%, jika dibandingkan jumlah epoch 100 didapatkan tingkat akurasinya sebesar 97,17%.

Hasil akurasi yang diperoleh untuk mengklasifikasikan gambar satwa liar dengan metode CNN yakni sebesar 97,17% dengan menggunakan skenario perbandingan data train validation 80% : 20%, ukuran kernel 3x3, optimizer RMSProp, 100 epoch, dan learning rate sebesar 0,001

**Keyword :** CNN, Klasifikasi Gambar, TNBBS, *Optimizer*, *Epoch*

## I. Pendahuluan

Taman Nasional Bukit Barisan Selatan (TNBBS) merupakan taman urutan ketiga terbesar di Pulau Sumatera yang memiliki keanekaragaman hayati yang tinggi. Kawasan TNBBS memiliki topografi dari 0 sampai 1964 m dpl dan luas wilayah sebesar 355.511 ha. TNBBS terletak di wilayah ujung selatan Provinsi Lampung sampai Provinsi Bengkulu bagian selatan [1]. TNBBS merupakan salah satu hutan konservasi warisan dunia yang memiliki ekosistem hutan hujan tropis yang lengkap. Selain itu, TNBBS juga memiliki kelompok spesies yang berperan dalam menjaga keanekaragaman hayati dan kestabilan ekosistem yaitu kucing liar [2]. Berdasarkan Undang- Undang No. 5 Tahun 1990 tentang Konservasi Sumber Daya Alam dan Ekosistem dan Peraturan Pemerintah No. 7 Tahun 1999 tentang Pengawetan Jenis Tumbuhan dan Satwa memiliki spesies kucing liar yang dilindungi di Indonesia. Beberapa

jenis spesies kucing liar antara lain harimau sumatera (*Panthera tigris sumatrae*), kucing merah (*Felis badia*), macan dahan (*Neofelis diardi*), kucing emas (*Catopuma temminckii*), kucing hidung pesek (*Prionailurus planiceps*), kucing bakau (*Prionailurus viverrinus*), kucing batu (*Pardofelis marmorata*), dan kucing congkok (*Prionailurus bengalensis*) [3].

Pada tahun 1997 TNBBS merupakan kawasan pelestarian alam yang memiliki kerjasama kegiatan konservasi dengan *Wildlife Conservation Society Indonesia Program* (WCS-IP). Tujuan dilakukan kerjasama tersebut adalah mendukung kegiatan penelitian ekologi kehidupan satwa liar di TNBBS. Pada setiap tahunnya dilaksanakan kegiatan tersebut, salah satunya adalah *Tropical Ecology Assessment and Monitoring* (TEAM). Kegiatan TEAM memiliki tujuan dalam pemantauan keberadaan satwa mamalia vertebrata di wilayah Resort Pemerihan-Way Haru dengan menggunakan

kamera pengawas [4]. Perkembangan teknologi yang secara pesat, kamera pengawas yang dipasang pada lokasi yang diinginkan akan menghasilkan rekaman keberadaan hewan secara otomatis yang menghasilkan data gambar.

Data gambar tersebut dihasilkan lebih relatif akurat serta dapat dipertanggung jawabkan [5]. Data gambar dapat memberikan sumber informasi terkait jenis spesies satwa langka dan jenis spesies satwa yang terancam. Data gambar tersebut dapat bertujuan mengetahui populasi satwa liar, hubungan ekologi dan dinamika populasi [6]. Selain itu, data gambar tersebut dapat diproses dan dianalisis data yang diperoleh di alam liar menggunakan metode *Convolutional Neural Networks* (CNN). Sebagian besar data gambar yang dihasilkan memiliki kualitas tingkat resolusinya tinggi sehingga diperlukan metode *Convolutional Neural Networks* (CNN). CNN adalah metode yang dapat mendeteksi objek dengan model yang dilatih sebelumnya pada dataset dengan skala yang sangat besar seperti ImageNet [7].

Penelitian sebelumnya telah menggunakan kamera pengawas dalam pemantauan satwa liar secara otomatis menggunakan metode *Convolutional Neural Networks*. Penelitian tersebut menghasilkan tingkat akurasi 96,6% untuk mendeteksi gambar satwa liar dan 90,4% untuk mengidentifikasi tiga spesies di antara kumpulan gambar satwa liar [8]. Sedangkan penelitian yang dilakukan oleh D. M. Blei dalam menggunakan algoritma klasifikasi gambar berbasis *traditional Bag-of-visual-words* dihasilkan tingkat akurasi sebesar 38,32% [9]. Sementara itu, penelitian lainnya yang dalam pembuatan model menggunakan metode CNN berupa dataset ImageNet, selanjutnya dataset tersebut dilatih ulang sehingga menghasilkan tingkat akurasi yang baik [7].

Berdasarkan uraian diatas, Penelitian ini akan dilakukan di lokasi TNBBS dengan memasang kamera pengawas sehingga data gambar dapat diklasifikasi menggunakan metode *Convolutional Neural Networks* (CNN). Dengan penelitian ini diharapkan dapat diketahui tingkat akurasi model yang diperoleh dan seberapa mampu model CNN mengklasifikasi gambar satwa liar dengan akurat.

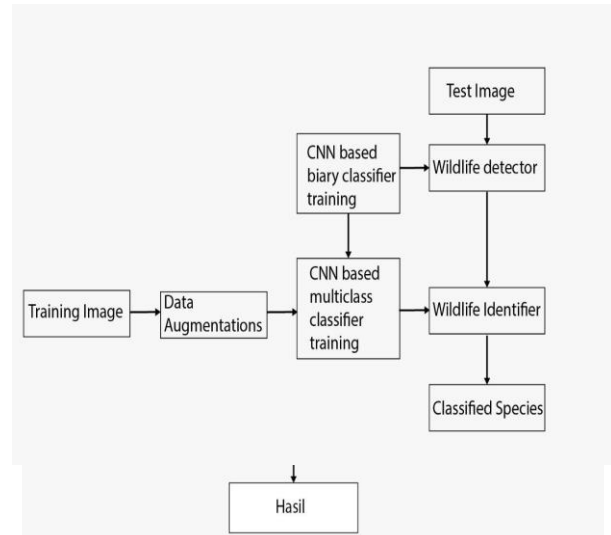
## II. Metode

Data yang digunakan dalam penelitian ini adalah data sekunder. Data diambil di situs [www.kaggle.com](http://www.kaggle.com). Penelitian ini menggunakan 3 kategori gambar hewan, yakni kucing, anjing, dan satwa liar.

Adapun diagram alir penelitian sebagai berikut.

**Gambar 2.1** Diagram Alir Penelitian

Adapun pembuatan model arsitektur CNN sebagai berikut.



**Gambar 2.2** Pemodelan Arsitektur CNN

Adapun tahapan penelitian sebagai berikut.

### 1. Pengumpulan data

Dalam tahap pengumpulan data, penelitian ini menggunakan data visual berupa gambar satwa liar yang terekam oleh kamera pengawas. Setelah didapatkan data tersebut maka dilakukan pengumpulan data satwa liar di TNBBS.

### 2. Pemrosesan data visual berupa gambar

Data yang dihasilkan berdasarkan hasil rekaman kamera pengawas memiliki noise atau memiliki resolusi tinggi sedangkan input model CNN harus dalam dimensi tetap. Oleh karena itu, dalam penelitian ditetapkan sebagai inputan gambar ukuran menjadi  $224 \times 224$  piksel dan gambar sudah dibuat label untuk pelatihan. Pada proses ini kami mengubah skala lebar dan tinggi gambar. Selain itu, intensitas piksel dinormalisasi ke dalam range sebesar  $[0,1]$ . Agar menghasilkan kualitas data yang baik maka dilakukan teknik augmentasi. Teknik augmentasi merupakan proses perubahan data gambar, menggeser dan zoom gambar yang dapat diterapkan pada gambar pelatihan tersebut. Selain itu data gambar dibagi menjadi dua yaitu data testing dan data training. Dalam penelitian ini digunakan data training 80% dan data testing 20%.

### 3. Membangun Neural Networks.

Penelitian ini membuat algoritma CNN dengan bahasa pemrograman python. Dalam hal ini, metode CNN dengan membuat dua model klasifikasi gambar berdasarkan data gambar yang didapatkan. Kedua model tersebut dirancang untuk melatih klasifikasi biner yaitu detektor satwa liar serta melatih kelas multi-kelas, yaitu pengidentifikasi satwa liar. Tiga arsitektur CNN yang biasa digunakan dalam penelitian adalah yaitu Lite AlexNet, VGG-16 dan ResNet-50. Namun, penelitian ini menggunakan Lite AlexNet. AlexNet Lite terdiri dari tiga lapisan konvolusional 2-D dengan aktivasi ULT dan MaxPooling. Dalam hal ini, kami menggunakan lapisan keluaran dengan aktivasi sigmoid untuk klasifikasi biner dan softmax untuk klasifikasi multikelas. Inputan pada arsitektur CNN berupa semua gambar berukuran tetap  $224 \times 224$  piksel.

### 4. Pelatihan *Deep Networks*

Dalam proses ini, menggunakan keras dan TensorFlow dengan optimizer Adam untuk pelatihan tersebut. Pada penelitian ini dilakukan pelatihan model CNN dalam bentuk set data.

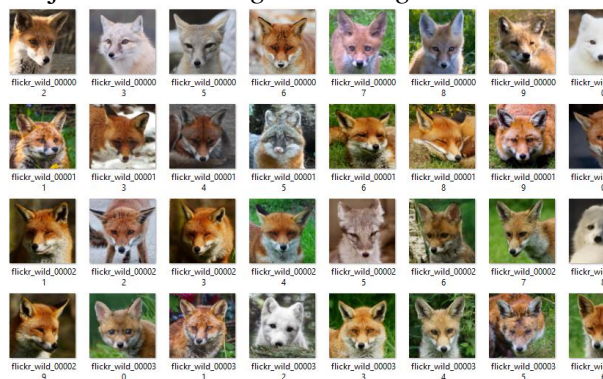
### 5. Pengujian

Tahap terakhir adalah pengujian data dengan menggunakan data gambar hewan tersebut.

## III. Hasil dan Pembahasan

### 3.1 Rancangan Pengujian

Penelitian ini menggunakan 3 kategori gambar hewan, yakni kucing, anjing, dan satwa liar. Dataset yang telah siap akan dibagi menjadi data *training* dan *testing*.



**Gambar 3.1** Pengelompokan Gambar hewan

Penelitian ini, pada setiap kategori menggunakan data sebanyak 3530 dari masing-masing jenis gambar hewan dimana perbandingan datanya yakni 80% : 20%. Perbandingan data tersebut didasarkan pada *pareto principle* yang umum digunakan dalam

*data mining*, dimana prinsip tersebut menyatakan bahwa 80% kejadian dihasilkan dari 20% sisanya. Maka total data *training* yang digunakan adalah  $(3530 \times 80) : 100 = 2824$ , sehingga data *validation* yang digunakan adalah sebesar  $3530 - 2824 = 706$ .

**Tabel 3.1** Skenario pembagian data

Data Training	Data validasi
80%	20%
2824	706

### 3.2 Merancang Arsitektur CNN

Model yang dibentuk menggunakan arsitektur CNN memiliki jumlah total parameter sebesar 1.087.619 dengan rincian sebagai berikut.

**Tabel 3.2** Perhitungan Parameter CNN

No.	Nama	Parameter
1	Conv2d_4 (conv 2D)	896
2	Batch_Normalization_5	128
3	Conv2d_5 (conv 2D)	9248
	Batch_Normalization_6	128
4	max_Pooling_2d_3 (MaxPooling 2D)	0
5	Conv2d_6 (conv 2D)	9248
6	Batch_Normalization_7	128
7	max_Pooling_2d_4 (MaxPooling 2D)	0
8	Conv2d_7 (conv 2D)	18496
9	Batch_Normalization_8	256
10	max_Pooling_2d_5 (MaxPooling 2D)	0
11	flatten_1 (Flatten)	0
..		
12	Dense_2 (Dense)	1048640
13	Batch_Normalization_9	256
14	Dropout_1 (Dropout)	0
15	Dense_3 (Dense)	195
	Total	1.087.619

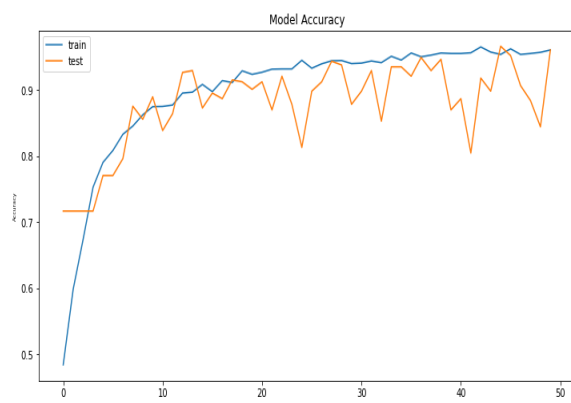
### 3.3 Model Hasil Training

Setelah melalui beberapa proses dalam algoritma *Convolutional Neural Network* (CNN) didapatkan hasil *training* dan *validation*. Proses ini menggunakan jumlah 50 epoch dan 100 epoch, nilai *learning rate* 0,001, dan *optimizer RMSprop*. Berikut hasil yang diperoleh berdasarkan parameter yang telah ditentukan:

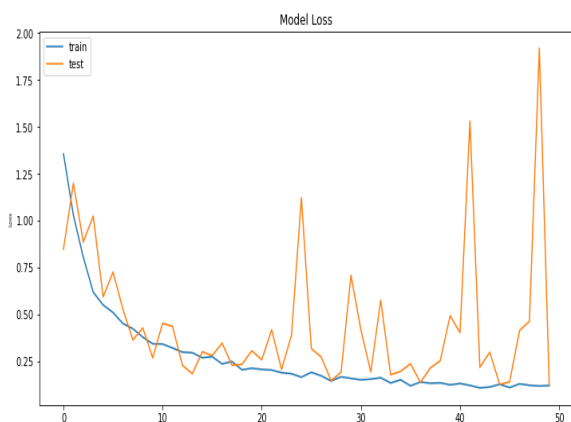
Classification Report				
	precision	recall	f1-score	support
cat	0.98	0.99	0.99	253
dog	0.87	0.92	0.89	50
wild	0.93	0.86	0.90	50
accuracy			0.96	353
macro avg	0.93	0.92	0.93	353
weighted avg	0.96	0.96	0.96	353

**Gambar 3. 2** Hasil Prediksi Optimizer RMSprop dengan Epoch sebesar 50

Berdasarkan gambar 3.2 dapat diketahui bahwa nilai akurasi dari *training* model dari masing-masing kelas adalah sebesar 98% untuk *cat*, 87% untuk *dog* dan 93% untuk *wild*. Sedangkan nilai *f1-score* dari masing-masing kelas adalah 99% untuk kucing, 89% untuk anjing dan 90% untuk satwa liar. Hasil pengujian terhadap performa akurasi dan loss dapat dilihat pada Gambar 3.3 dan Gambar 3.4.



**Gambar 3.3** Grafik Akurasi model CNN dengan optimizer RMSprop Epoch 50.

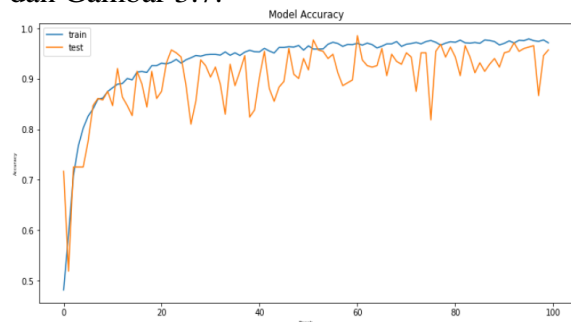


**Gambar 3.4** Grafik Loss model CNN dengan optimizer RMSprop Epoch 50 .

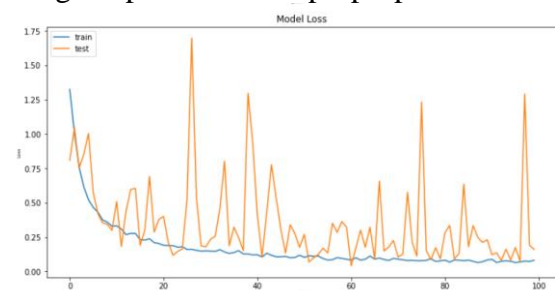
Classification Report				
	precision	recall	f1-score	support
cat	0.97	1.00	0.98	253
dog	0.66	0.94	0.78	50
wild	0.95	0.42	0.58	50
accuracy			0.91	353
macro avg	0.86	0.79	0.78	353
weighted avg	0.92	0.91	0.90	353

**Gambar 3. 5** Hasil Prediksi Optimizer RMSprop dengan Epoch sebesar 100

Berdasarkan gambar 3.5 dapat diketahui bahwa nilai akurasi dari *training* model dari masing-masing kelas adalah sebesar 97% untuk kucing, 66% untuk anjing dan 95% untuk satwa liar. Sedangkan nilai *f1-score* dari masing-masing kelas adalah 98% untuk kucing, 78% untuk anjing dan 58% untuk satwa liar. Hasil pengujian terhadap performa akurasi dan loss dapat dilihat pada Gambar 3.6 dan Gambar 3.7.



**Gambar 3. 6** Grafik Akurasi model CNN dengan optimizer RMSprop Epoch 100.



**Gambar 3. 7** Grafik Loss model CNN dengan optimizer RMSprop Epoch 100

Sedangkan berdasarkan gambar 3.9 nilai-nilai yang dihasilkan dari keluaran akhir *epoch* 100 Optimizer RMSprop adalah akurasi sebesar 97,17%, validation loss sebesar 16,10%, dan *validation accuracy* sebesar 95,75%. Melalui hasil *epoch* berikut dapat diketahui bahwa terdapat korelasi antar nilai akurasi dan nilai *loss* pada data *train* dengan banyaknya *epoch* atau iterasi. Semakin besar *epoch* yang digunakan, maka nilai akurasi pada data train semakin tinggi. Berbanding terbalik dengan



nilai akurasi, semakin besar epoch yang digunakan maka nilai *loss* yang dihasilkan pada pelatihan data semakin rendah. Berdasarkan hal tersebut, maka dapat disimpulkan bahwa untuk memperkecil nilai *loss* yang didapatkan maka dapat dilakukan dengan cara memperbanyak jumlah *epoch* pada proses *training*, sehingga model akan menghasilkan nilai akurasi yang lebih tinggi. Sedangkan, tidak terdapat korelasi atau hubungan dengan banyaknya *epoch* untuk nilai akurasi dan nilai *loss* pada *data test*.

```
Epoch 94/100
57/57 [=====] - 299s 5s/step - loss: 0.0705 - accuracy: 0.9766 - val_loss: 0.1628 - val_accu
Epoch 95/100
57/57 [=====] - 303s 5s/step - loss: 0.0747 - accuracy: 0.9759 - val_loss: 0.0842 - val_accu
Epoch 96/100
57/57 [=====] - 300s 5s/step - loss: 0.0654 - accuracy: 0.9795 - val_loss: 0.1750 - val_accu
Epoch 97/100
57/57 [=====] - 303s 5s/step - loss: 0.0700 - accuracy: 0.9759 - val_loss: 0.0756 - val_accu
Epoch 98/100
57/57 [=====] - 302s 5s/step - loss: 0.0761 - accuracy: 0.9745 - val_loss: 1.2913 - val_accu
Epoch 99/100
57/57 [=====] - 302s 5s/step - loss: 0.0733 - accuracy: 0.9773 - val_loss: 0.1908 - val_accu
Epoch 100/100
57/57 [=====] - 305s 5s/step - loss: 0.0819 - accuracy: 0.9717 - val_loss: 0.1610 - val_accu
```

**Gambar 3. 9** Hasil Perhitungan Optimizer RMSprop dengan Epoch 100

### 3.4 Perbandingan Epoch dengan Optimizer

**Tabel 3. 3** Perbandingan Epoch dengan Optimizer Adam

Epoch	Accuracy Validaton	Loss Validation
10	0.8357	0.4116
20	0.8754	0.4465
30	0.9377	0.1611
40	0.9122	0.3054
50	0.9008	0.2883
60	0.9235	0.2982
70	0.9575	0.1086
80	0.9490	0.2173
90	0.9660	0.1077
100	0.9065	0.2797

Berdasarkan iterasi *epoch* yang dihasilkan dapat diketahui bahwa dengan menggunakan *optimizer* Adam, *epoch* sebesar 100

menghasilkan nilai *accuracy validation* yaitu 0,9065 atau 90,65%. Namun nilai *loss validation* yang dihasilkan memiliki nilai *loss* yang sedikit rendah sehingga dapat dikatakan bahwa diantara sepuluh percobaan, jumlah *epoch* 100 adalah yang terbaik.

**Tabel 3. 4** Perbandingan Epoch dengan Optimizer RMSProp

Epoch	Accuracy Validaton	Loss Validation
10	0.8754	0.3001
20	0.8612	0.3790
30	0.9263	0.1790
40	0.8385	0.9381
50	0.9405	0.1760
60	0.8980	0.3241
70	0.9292	0.2261
80	0.9632	0.0928
90	0.9235	0.2310
100	0.9575	0.1610

Berdasarkan iterasi *epoch* yang dihasilkan dapat diketahui bahwa dengan menggunakan *optimizer* RMSprop, *epoch* sebesar 100 menghasilkan nilai *accuracy validation* yaitu 0,9575 atau 95,75%. Namun nilai *loss validation* yang dihasilkan memiliki nilai *loss* yang sedikit rendah sehingga dapat dikatakan bahwa diantara sepuluh percobaan, jumlah *epoch* 100 adalah yang terbaik

```
Classification Report
              precision    recall  f1-score   support

   cat         0.92         1.00         0.96         253
   dog         0.95         0.76         0.84          50
  wild         1.00         0.78         0.88          50

 accuracy                   0.93         353
 macro avg         0.96         0.85         0.89         353
 weighted avg         0.94         0.93         0.93         353
```

**Gambar 3. 10** Hasil Akurasi Model dengan Optimizer Adam dan Epoch 50

Gambar 3.10 merupakan hasil dari akurasi model CNN menggunakan *epoch* 50 dengan *optimizer* Adam, dimana akurasi dari masing-masing kategori adalah 96% untuk kucing, 84% untuk anjing dan 88% untuk satwa liar.

Classification Report				
	precision	recall	f1-score	support
cat	0.98	0.99	0.99	253
dog	0.87	0.92	0.89	50
wild	0.93	0.86	0.90	50
accuracy			0.96	353
macro avg	0.93	0.92	0.93	353
weighted avg	0.96	0.96	0.96	353

**Gambar 3. 11** Hasil Akurasi Model dengan Optimizer RMSProp dan Epoch 50  
 Gambar 3.11 merupakan *output* dari nilai akurasi dengan menggunakan *epoch* 50 dengan Optimizer RMSProp. Dapat diketahui bahwa seluruh nilai yang dihasilkan pada kucing, anjing, dan satwa liar akurasi yang dihasilkan adalah sebesar 96%.

Classification Report				
	precision	recall	f1-score	support
cat	0.97	1.00	0.98	253
dog	0.66	0.94	0.78	50
wild	0.95	0.42	0.58	50
accuracy			0.91	353
macro avg	0.86	0.79	0.78	353
weighted avg	0.92	0.91	0.90	353

**Gambar 3. 12** Hasil Akurasi Model dengan Optimizer Adam dan Epoch 100

Gambar 3.12 merupakan hasil dari akurasi model CNN menggunakan *epoch* 100 dengan Optimizer Adam dimana akurasi dari masing-masing kategori adalah 98% untuk kucing, 78% untuk anjing dan 58% untuk satwa liar.

Classification Report				
	precision	recall	f1-score	support
cat	0.95	1.00	0.97	253
dog	1.00	0.92	0.96	50
wild	0.97	0.78	0.87	50
accuracy			0.96	353
macro avg	0.97	0.90	0.93	353
weighted avg	0.96	0.96	0.96	353

**Gambar 3. 13** Hasil Akurasi Model dengan Optimizer RMSProp dan Epoch 100

Gambar 3.13 merupakan *output* dari nilai akurasi dengan menggunakan *epoch* 100 dengan Optimizer RMSProp. Dapat diketahui bahwa seluruh nilai yang dihasilkan pada kucing 97%, anjing 96%, dan satwa liar akurasi yang dihasilkan adalah sebesar 87%.

### 3.5 Hasil Klasifikasi Model Terbaik

Pada tahap ini akan dilakukan perbandingan untuk menentukan model terbaik dari keseluruhan model yang telah dibuat. Berikut adalah tabel dari masing-

masing model terbaik berdasarkan *optimizer* dan *epoch*-nya.

**Tabel 3. 4 Perbandingan Optimizer**

Optimizer	Accuracy Validation	Loss Validation	Accuracy
Adam 50	0.9008	0.2883	95,22%
RMSProp 50	0.9405	0.1760	96,03%
Adam 100	0.9065	0.2797	96,74%
RMSProp 100	0.9575	0.1610	97,17%

Berdasarkan perbandingan dari tabel diatas, dapat diketahui bahwa *optimizer* RMSProp dengan *epoch* sebanyak 100 dapat menghasilkan nilai akurasi yang lebih tinggi dibandingkan dengan *optimizer* Adam. Hasil akurasi *optimizer* Adam dan RMSprop tidak jauh berbeda, karena Adam merupakan kombinasi antara RMSprop dengan momentum.

### Kesimpulan

Berdasarkan analisis yang telah dilakukan, diperoleh kesimpulan antara lain:

1. Pada klasifikasi gambar satwa liar menggunakan metode CNN dimana arsitektur terbaik didapatkan dengan perbandingan 2 parameter yakni epoch dan optimizer. Diperoleh perbandingan akurasi dari 2 jenis optimizer yaitu 95,22% untuk optimizer Adam pada epoch sebesar 50 sedangkan pada epoch 100 didapatkan akurasi 96,74%. Pada optimizer RMSProp dengan epoch 50 epoch dihasilkan 96,03%, jika dibandingkan jumlah epoch 100 didapatkan tingkat akurasinya sebesar 97,17%.
2. Hasil akurasi yang diperoleh untuk mengklasifikasikan gambar satwa liar dengan metode CNN yakni sebesar 97,17% dengan menggunakan skenario perbandingan data train validation 80% : 20%, ukuran kernel 3x3, optimizer RMSProp, 100 epoch, dan learning rate sebesar 0,001

## IV. Referensi

1. Taman Nasional Bukit Barisan Selatan (TNBBS).2017.Diakses pada tanggal 5 Juni 2021,10.00 WIB.  
<<https://www.tnbbs.org>>
2. Miller, B., D. Foreman, C.M. del Rio, R. Noss, M. Philips, R. Reading,
3. M.E. Soule, J. Terborgh & L. Wilcox. 2001. The importance of large carnivores to healthy ecosystem. *Endangered Species UPDATE* 18(5): 202-210
4. Sunarto, M.J. Kelly, S. Klenzendorf, M.R.Vaughan, Zulfahmi, M.B. Hutajulu and K. Parakkasi. 2013. Threatened predator on the equator: multi-point abundance estimates of the tiger *Panthera tigris* in central Sumatra. *Oryx* 47(2):211-220
5. [WCS-IP]. (2015). Laporan Tahunan Sumatera Program Wildlife Conservation Society Indonesia Program.Wibisono, H.T. and J. Maccarthy. 2010. Melanistic marbled cat from Bukit Barisan Selatan National Park, Sumatra, Indonesia. *Cat News* 52:9-1
6. A. F. O'Connell, J. D. Nichols, and K. U. Karanth, *Camera traps in animal ecology: Methods and Analyses*. Springer Science & Business Media, 2010.
7. A. Gómez, A. Salazar, and F. Vargas, "Towards automatic wild animal monitoring: Identification of animal species in camera-trap images using very deep convolutional neural networks," *arXiv:1603.06169*, 2016.
8. Nguyen. Hung, Sarah J. Maclagan, Tu Dinh Nguye.2017. Animal recognition and identification deep convolutional neural networks for Automated Wildlife Monitoring
9. D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.

