

Muhammad Andhika Ramadhan

AlphaZero

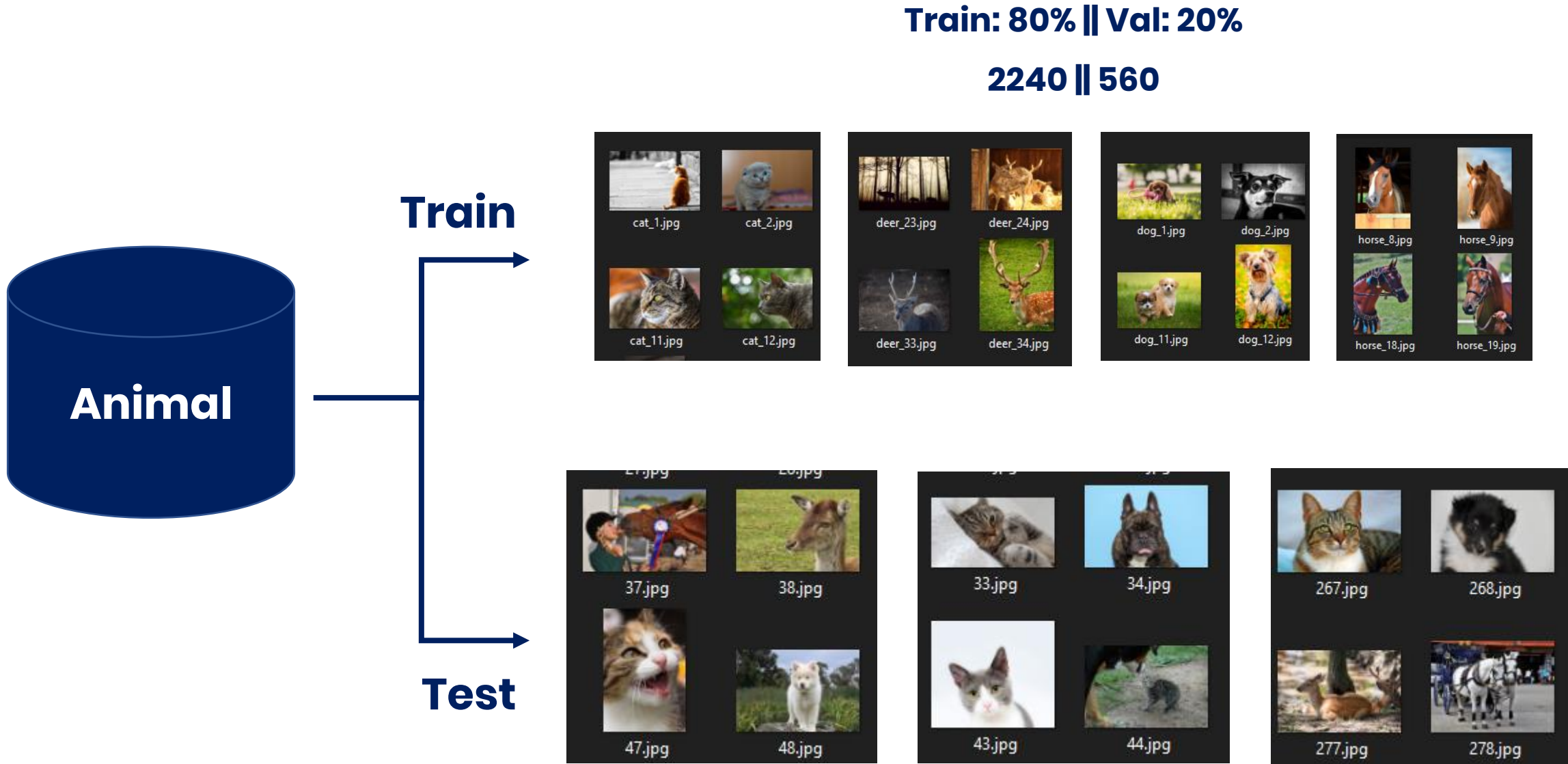
Animal Image Classification Using EfficientNet

Ujian Praktik Computer Vision – Orbit Future Academy

Latar Belakang dan Rumusan Masalah

- **Latar Belakang**
 - Terdapat 4 jenis hewan yang akan di klasifikasi
 - 4 hewan tersebut antara lain: Kucing, Rusa, Anjing dan Kuda
 - Terdapat 2800 gambar keseluruhan untuk train
 - 729 gambar untuk testing
- **Rumusan Masalah**
 - Membuat sebuah arsitektur untuk mengklasifikasikan jenis hewan apa yang ada pada gambar

Data, Variabel yang digunakan



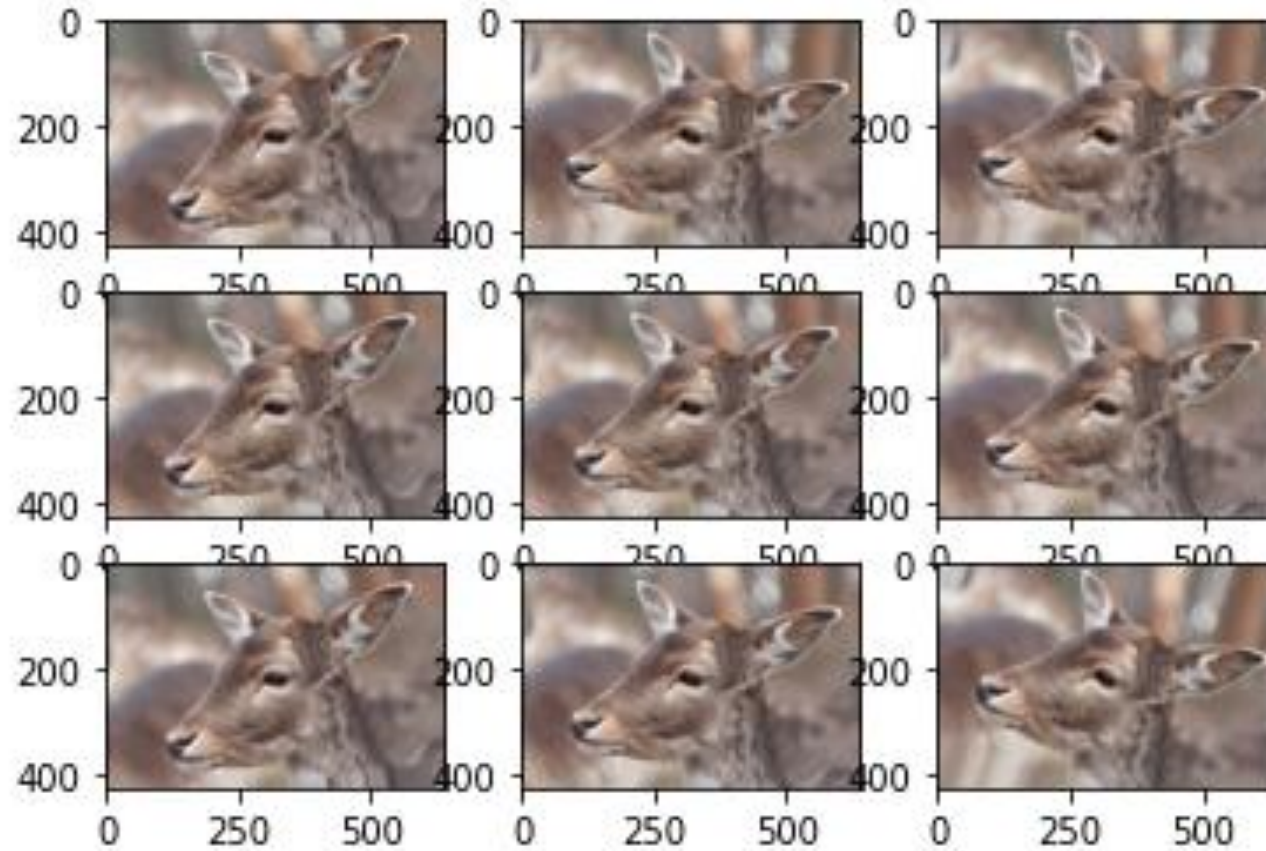
Preprocessing

```
image_datagen = ImageDataGenerator(  
#           rescale=1./255,  
           rotation_range=20,  
           horizontal_flip=True,  
           shear_range = 0.2,  
           fill_mode = 'nearest',  
           validation_split = 0.2,  
           width_shift_range=0.1,  
           height_shift_range=0.1,)
```

<https://machinelearningmastery.com/how-to-configure-image-data-augmentation-when-training-deep-learning-neural-networks/>

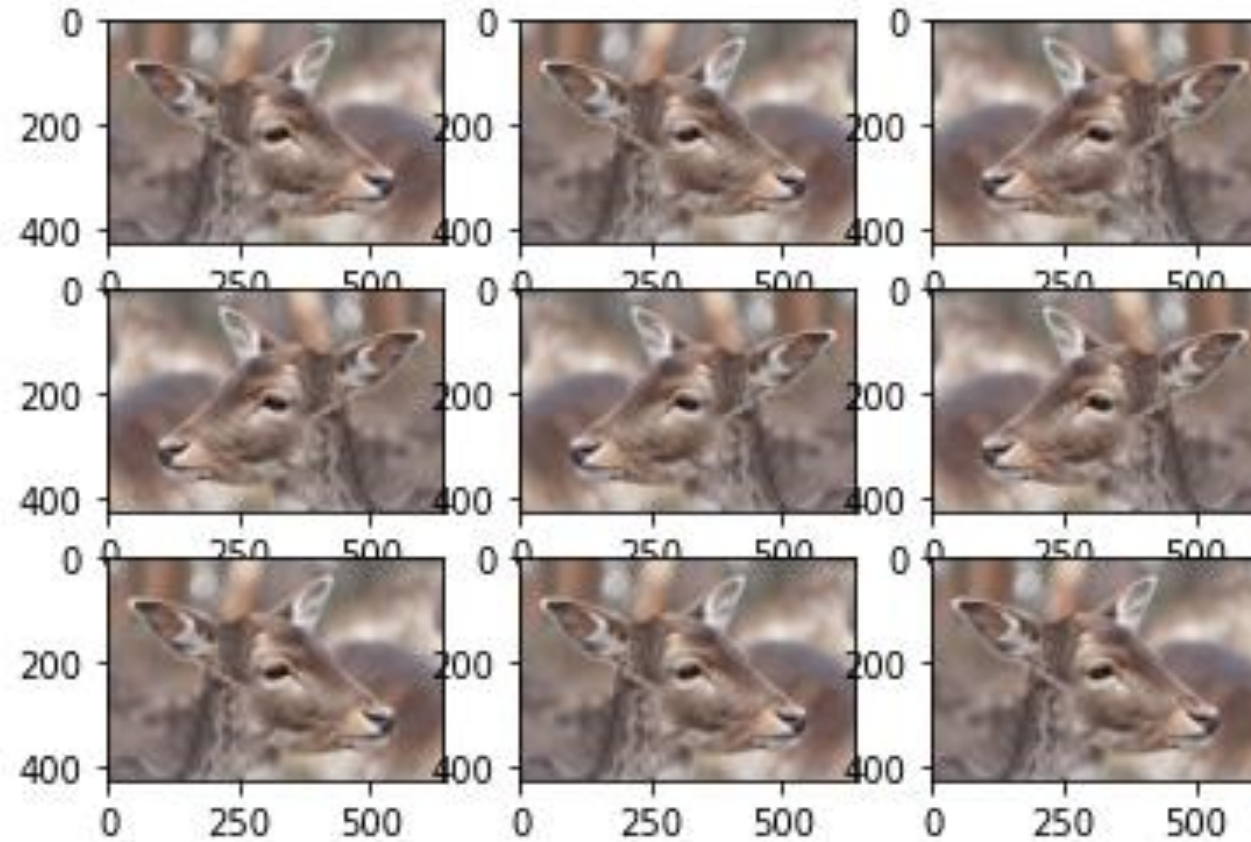
Preprocessing (Rotation Range)

rotation_range = 20



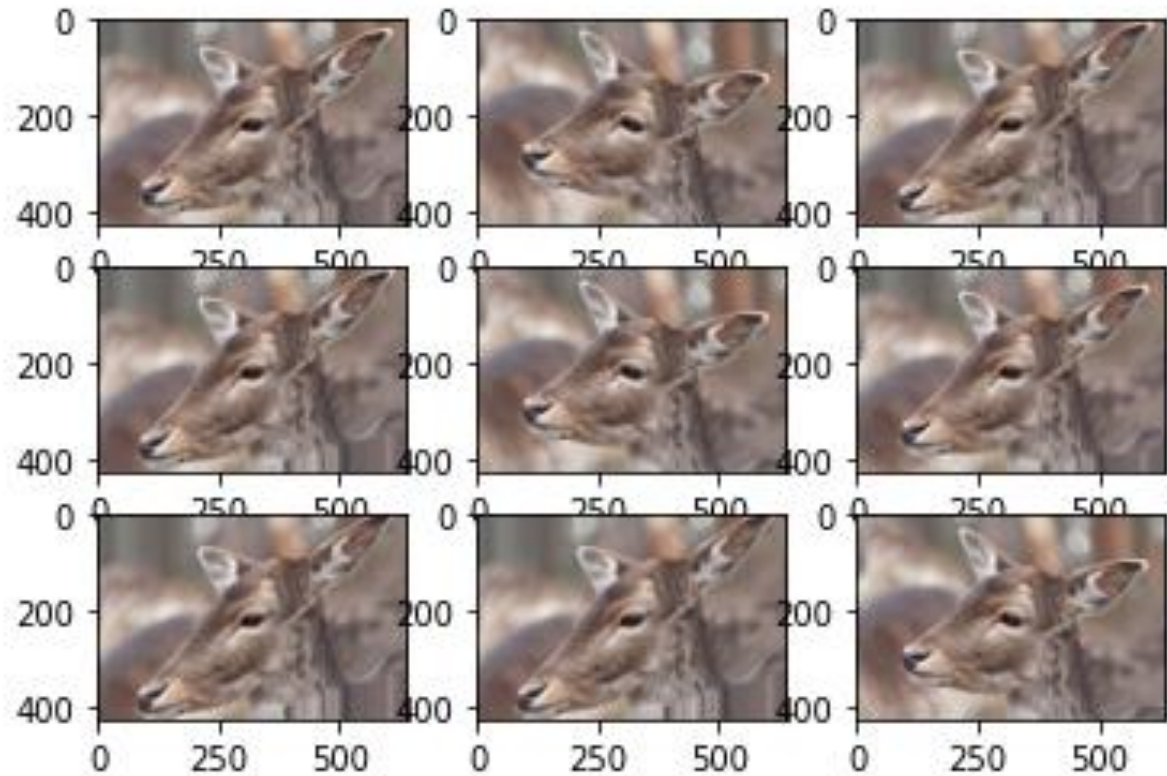
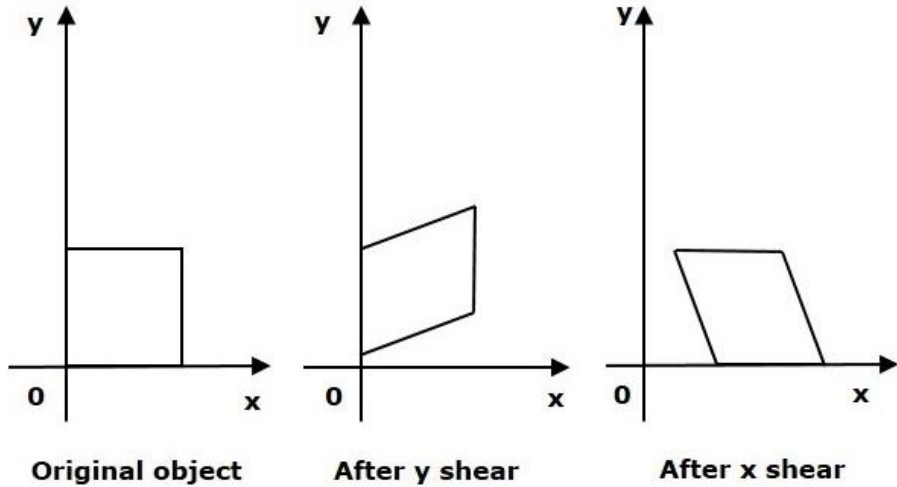
Preprocessing (Horizontal Flip)

horizontal_flip = True



Preprocessing (Shear Range)

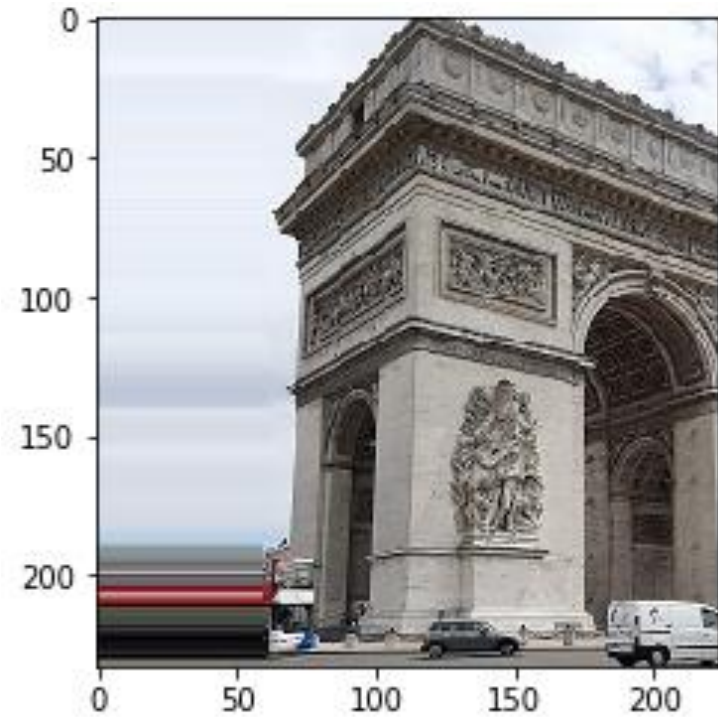
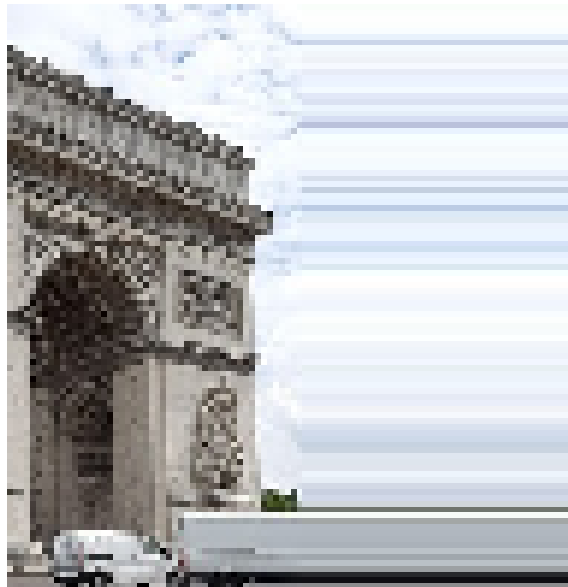
shear_range = 20



Preprocessing (Fill Mode)

`fill_mode = 'nearest'`

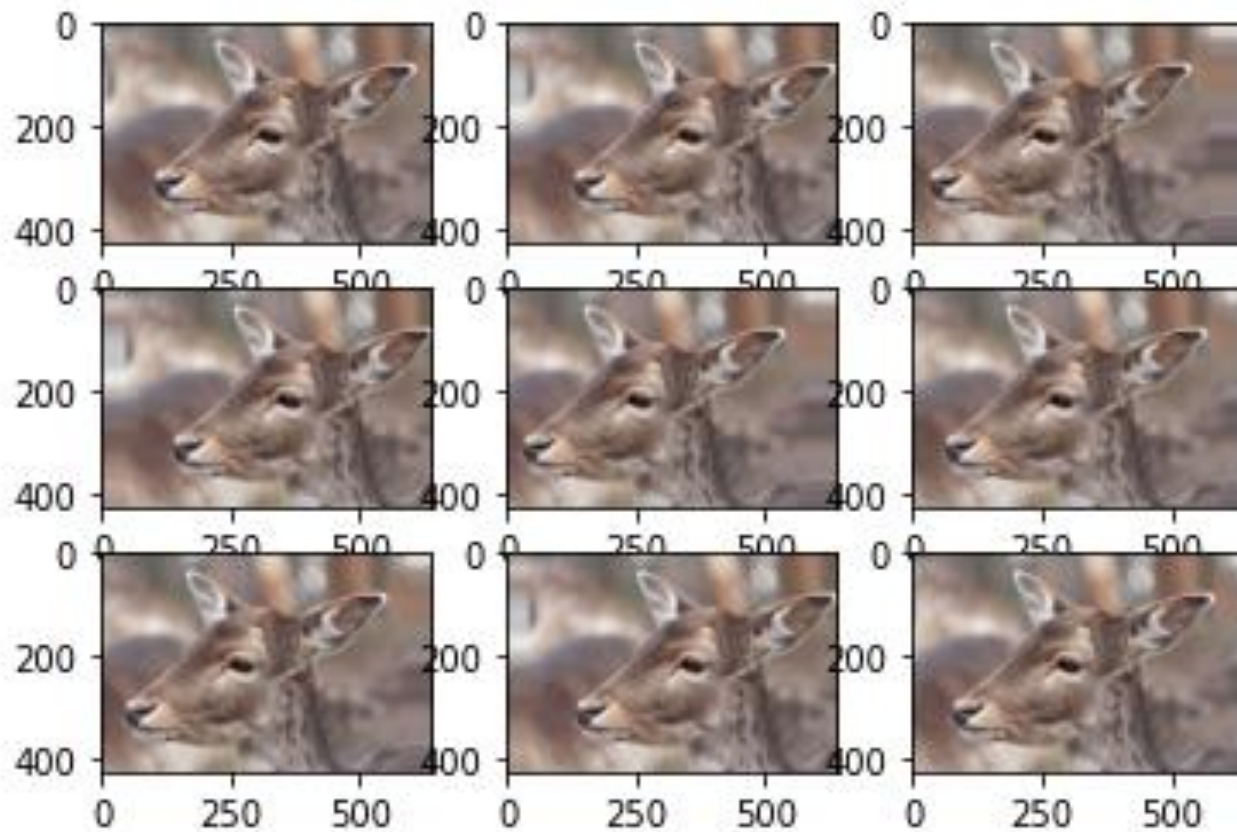
NEAREST



<https://towardsdatascience.com/complete-image-augmentation-in-opencv-31a6b02694f5>

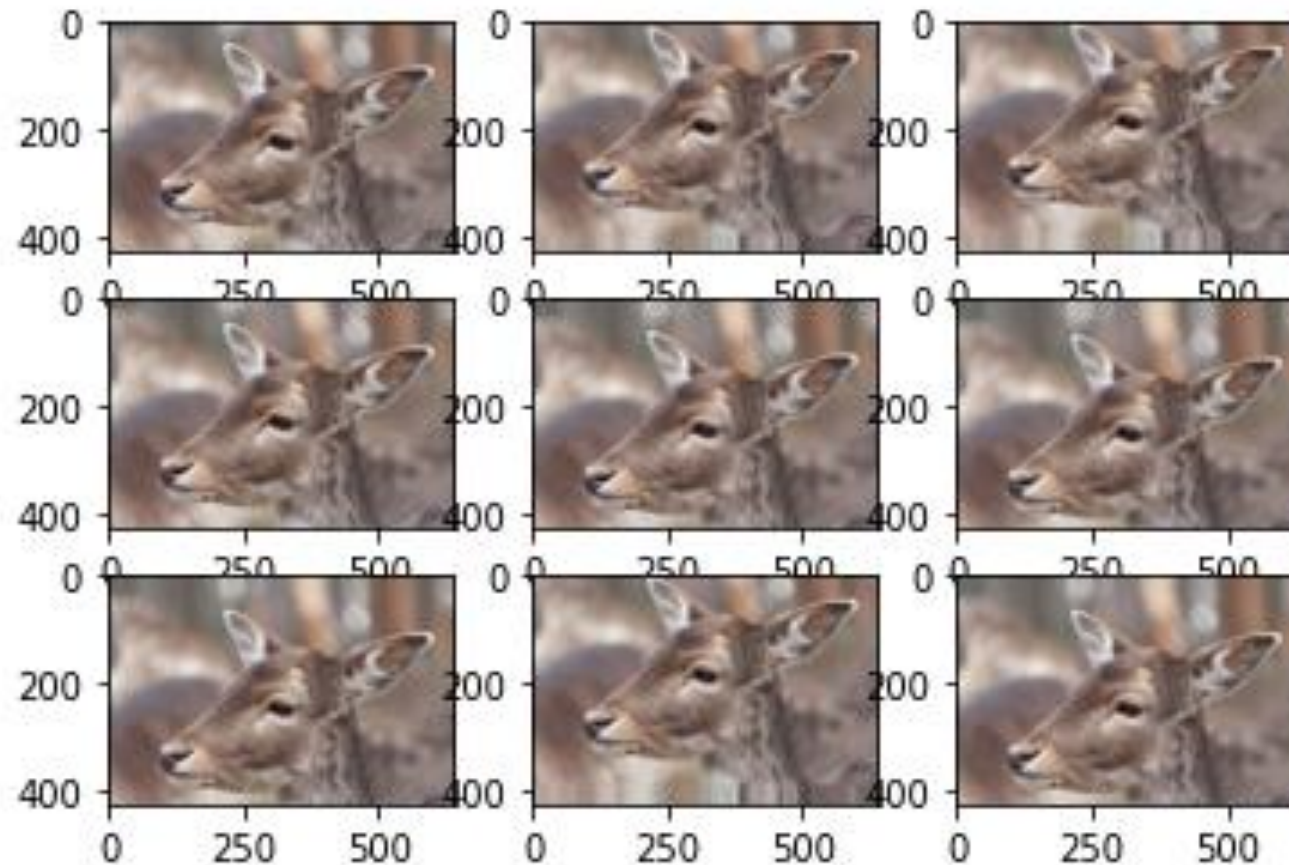
Preprocessing (Width Shift Range)

width_shift_range = 0.1



Preprocessing (Height Shift Range)

height_shift_range = 0.1



Model dan Parameter

Default

Model: "sequential_2"

Layer (type)	Output Shape	Param #
conv2d_10 (Conv2D)	(None, 298, 298, 32)	896
max_pooling2d_10 (MaxPooling2D)	(None, 149, 149, 32)	0
conv2d_11 (Conv2D)	(None, 147, 147, 64)	18496
max_pooling2d_11 (MaxPooling2D)	(None, 73, 73, 64)	0
conv2d_12 (Conv2D)	(None, 71, 71, 128)	73856
max_pooling2d_12 (MaxPooling2D)	(None, 35, 35, 128)	0
conv2d_13 (Conv2D)	(None, 33, 33, 128)	147584
max_pooling2d_13 (MaxPooling2D)	(None, 16, 16, 128)	0
conv2d_14 (Conv2D)	(None, 14, 14, 512)	590336
max_pooling2d_14 (MaxPooling2D)	(None, 7, 7, 512)	0
flatten_1 (Flatten)	(None, 25088)	0
dense_4 (Dense)	(None, 512)	12845568
dense_5 (Dense)	(None, 4)	2052
Total params: 13,678,788		
Trainable params: 13,678,788		
Non-trainable params: 0		

Epoch 1/20

Epoch 1: val_accuracy improved from -inf to 0.32143, saving model to checkpoints/best.h5
70/70 - 71s - loss: 1.9573 - accuracy: 0.3076 - val_loss: 1.3791 - val_accuracy: 0.3214 - 71s/epoch - 1s/step
Epoch 2/20

Epoch 2: val_accuracy improved from 0.32143 to 0.32857, saving model to checkpoints/best.h5
70/70 - 69s - loss: 1.3745 - accuracy: 0.3210 - val_loss: 1.3707 - val_accuracy: 0.3286 - 69s/epoch - 993ms/step
Epoch 3/20

Epoch 3: val_accuracy improved from 0.32857 to 0.33393, saving model to checkpoints/best.h5
70/70 - 69s - loss: 1.3667 - accuracy: 0.3379 - val_loss: 1.3627 - val_accuracy: 0.3339 - 69s/epoch - 984ms/step
Epoch 4/20

Model dan Parameter

EfficientNetB3

Model: "sequential_9"

Layer (type)	Output Shape	Param #
efficientnetb3 (Functional)	(None, 10, 10, 1536)	10783535
global_average_pooling2d_9 ((None, 1536)	0
flatten_9 (Flatten)	(None, 1536)	0
dense_27 (Dense)	(None, 1280)	1967360
dense_28 (Dense)	(None, 128)	163968
dense_29 (Dense)	(None, 4)	516
Total params: 12,915,379		
Trainable params: 12,828,076		
Non-trainable params: 87,303		

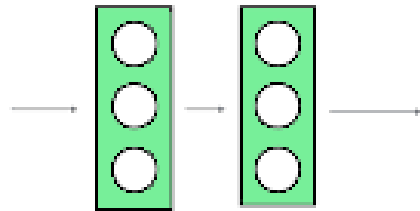
Block NO. (i)	Layer (F_i ())	Resolution ($H_i \times W_i$)	No. of Layers (L_i)
1	Conv 3x3	300x300	1
2	MBConv1, 3x3	150x150	2
3	MBConv6, 3x3	150x150	3
4	MBConv6, 5x5	75x75	3
5	MBConv6, 3x3	38x38	5
6	MBConv6, 5x5	19x19	5
7	MBConv6, 5x5	10x10	6
8	MBConv6, 3x3	10x10	2
9	Conv 1x1	10x10	1
10	Global Pooling	10x10	1
11	Dense layer	10x10	1

Model dan Parameter

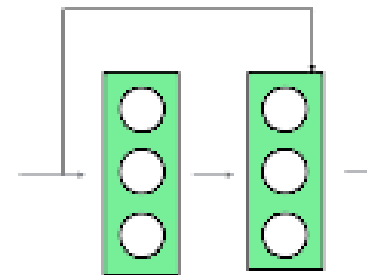
MBConv?

- **Blok Residual Terbalik yang menghemat waktu dan memori dengan mengganti konvolusi 3 X 3 dengan konvolusi yang lebih mendalam.**
- **Dicetus mobilenet untuk meningkatkan efisiensi**
- **Residual Block merupakan suatu kumpulan fungsi yang didalamnya terdapat fitur Skip Connection**

without skip connection



with skip connection



<https://medium.com/python-in-plain-english/implementing-efficientnet-in-pytorch-part-3-mbconv-squeeze-and-excitation-and-more-4ca9fd62d302>

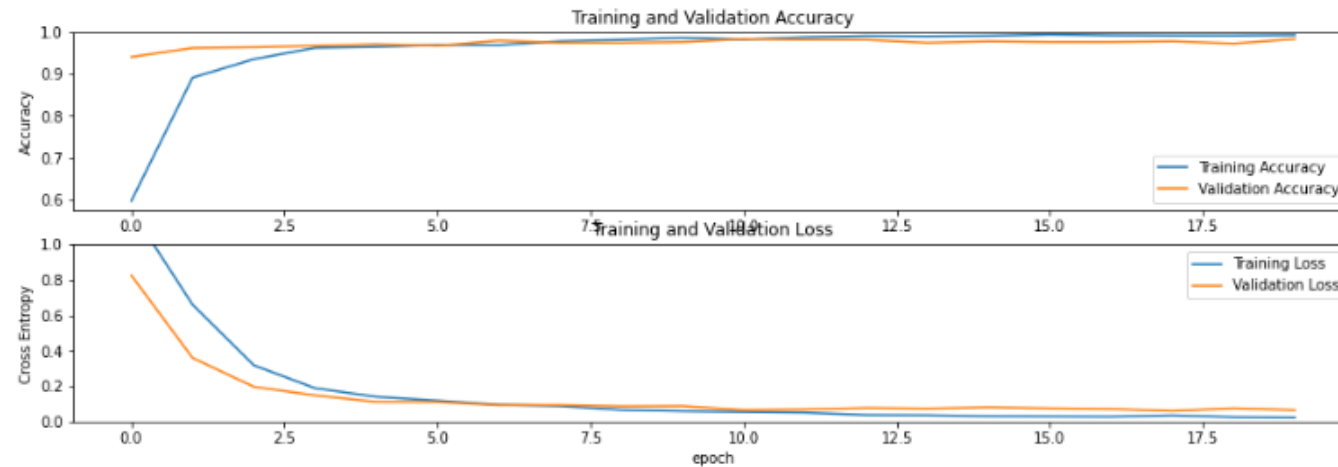
Model dan Parameter

Parameter	Value
Optimizer	SGD
Learning Rate	0.0010
Momentum	0.9
Loss	Categorical_crossentropy
Metrics	accuracy

Ukuran Kebaikan Model

```
results = model.evaluate(validation_generator)
print(f"test loss {results[0]}, test accuracy {results[1]}")
```

18/18 [=====] - 15s 819ms/step - loss: 0.0650 - accuracy: 0.9804
test loss 0.06495903432369232, test accuracy 0.9803571701049805









submission3.csv
a day ago by mdhkrmd


Ver 3

0.98353



Leaderboard

#	Team	Members	Score	Entries	Last	Code
1	Naufal Al Hakim		0.99176	10	9h	
2	JeffKing		0.98628	5	10d	
3	Prashanth-ACSQ		0.98628	13	2d	
4	Dastin Aryo		0.98628	2	1d	
5	Erni Nuraini		0.98491	5	10h	
6	mdhkrmd		0.98353	7	16h	



Your Best Entry!
Your most recent submission scored 0.98353, which is the same as your previous score. Keep trying!

Kesimpulan

- **Transfer Learning sukses menaikan akurasi**
- **Akurasi yang dihasilkan mencapai 0.98353 (Kaggle)**
- **EfficientNet, dengan parameter yang tidak terlalu besar, namun menghasilkan akurasi yang baik**

Terima Kasih ☺

Ujian Praktik – Orbit Future Academy