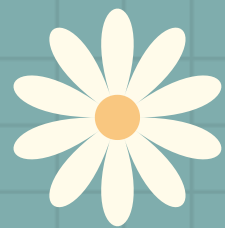




● PROYEK ●



Klasifikasi Gambar dengan Machine Learning



oleh Selviana

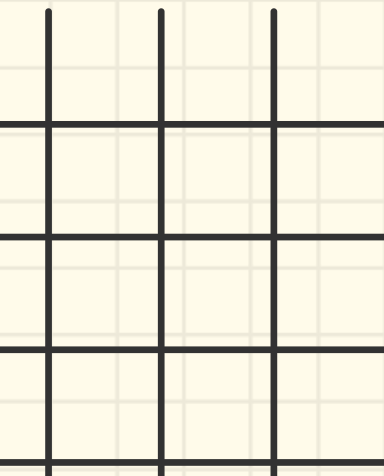




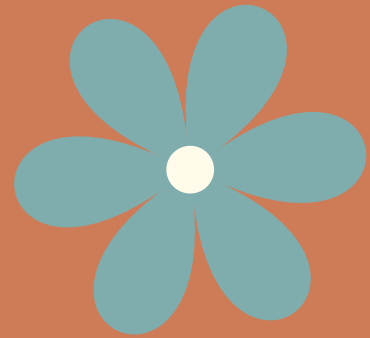
Introduction

Proyek Klasifikasi Gambar dengan menggunakan salah satu metode Machine Learning yaitu Convolution Neural Network (CNN) adalah proyek akhir untuk kelas Belajar Machine Learning untuk Pemula.

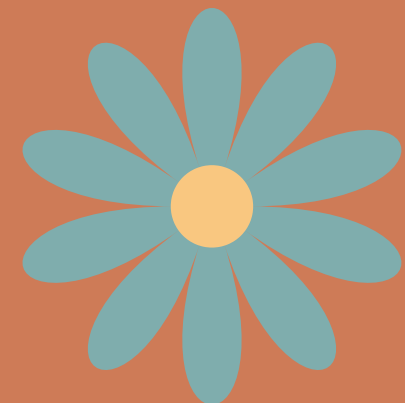
Kelas tersebut merupakan bagian dari beasiswa Lintasarta Cloudeka Digischool 2023 x Dicoding Indonesia.




Tujuan Proyek



Tujuan dari proyek ini untuk mengenali gambar tangan yang membentuk gunting, batu dan kertas. Program dibuat dengan jaringan syaraf tiruan (Convolution Neural Network (CNN) menggunakan Tensorflow.





TOOLS



Google Colaboratory

Bahasa Pemrograman Python





Tahapan Pemrograman



01

Persiapan Dataset

Perancangan Model

03

02

Proses Augmentasi

Pelatihan Model

04

05

Pengujian Model



01

Persiapan Daset

Library yang digunakan

```
✓ [1] import tensorflow as tf  
5s from tensorflow.keras.optimizers import RMSprop  
from tensorflow.keras.preprocessing.image import ImageDataGenerator  
  
✓ [2] import os  
0s import numpy as np  
from google.colab import files  
from keras.preprocessing import image  
import matplotlib.pyplot as plt  
import matplotlib.image as mpimg  
%matplotlib inline
```

Pengunduhan Dataset

```
[3] !wget --no-check-certificate \  
https://github.com/dicodingacademy/assets/releases/download/release/rockpaperscissors.zip
```

Pendefinisian Direktori

```
[7] import zipfile,os  
local_zip = '/content/rockpaperscissors.zip'  
zip_ref = zipfile.ZipFile(local_zip, 'r')  
zip_ref.extractall('/content')  
zip_ref.close()  
  
[8] base_dir = '/content/rockpaperscissors/rps-cv-images'  
  
[9] os.listdir(base_dir)  
  
['paper', 'scissors', 'rock', 'README_rpc-cv-images.txt']
```

02

Proses Augmentasi

Pembuatan data Generator

```
[8] Generator = ImageDataGenerator(  
    rescale = 1/255.0,  
    rotation_range = 20,  
    horizontal_flip = True,  
    shear_range=0.2,  
    fill_mode = 'nearest',  
    validation_split=0.4  
)
```

Pembuatan data Generator untuk data training dan data validation

```
▶ train_gen = Generator.flow_from_directory(  
    base_dir,  
    target_size=(150,150),  
    batch_size=32,  
    class_mode='categorical',  
    subset='training'  
)  
  
validation_gen = Generator.flow_from_directory(  
    base_dir,  
    target_size=(150,150),  
    batch_size=32,  
    class_mode='categorical',  
    subset='validation'  
)
```

```
Found 1314 images belonging to 3 classes.  
Found 874 images belonging to 3 classes.
```

03

Perancangan Model

Pembuatan Model Sequential

```
[11] model = tf.keras.models.Sequential([
    tf.keras.layers.Conv2D(32, (3,3), activation='relu', input_shape=(150, 150, 3)),
    tf.keras.layers.MaxPooling2D(2, 2),
    tf.keras.layers.Conv2D(64, (3,3), activation='relu'),
    tf.keras.layers.MaxPooling2D(2,2),
    tf.keras.layers.Conv2D(128, (3,3), activation='relu'),
    tf.keras.layers.MaxPooling2D(2,2),
    tf.keras.layers.Conv2D(128, (3,3), activation='relu'),
    tf.keras.layers.MaxPooling2D(2,2),
    tf.keras.layers.Flatten(),
    tf.keras.layers.Dense(512, activation='relu'),
    tf.keras.layers.Dense(3, activation='softmax')
])
```

Optimizer untuk Model

```
[13] model.compile(loss='categorical_crossentropy',
                    optimizer=RMSprop(),
                    metrics=['accuracy'])
```


04

Pelatihan Model

Proses Pelatihan Model

```
[14] ModelLatih = model.fit(  
    train_gen,  
    steps_per_epoch = 25,  
    epochs = 20,  
    validation_data = validation_gen,  
    validation_steps = 5,  
    verbose = 2,  
)
```

Evaluasi Model

```
val_loss: 0.0906 - val_accuracy: 0.9750
```

05

Pengujian model

Proses Pengujian Model

```
uploaded = files.upload()

for fn in uploaded.keys():

    # predicting images
    path = fn
    img = image.load_img(path, target_size=(150,150))
    x = image.img_to_array(img)
    x = np.expand_dims(x, axis=0)
    images = np.vstack([x])

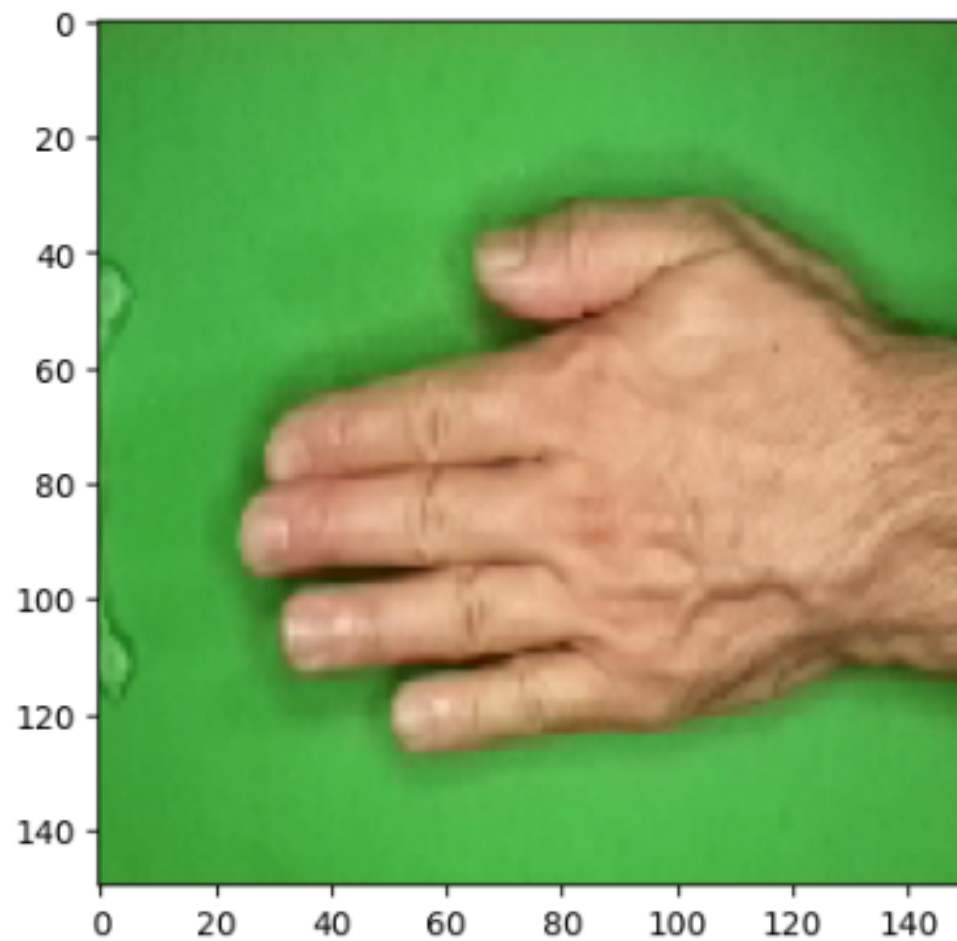
    plt.imshow(img)
    plt.show

    classes = model.predict(images, batch_size=10)
    print(fn)
    if classes[0][0] == 1:
        print('Gambar ini adalah Paper')
    elif classes[0][1] == 1:
        print('Gambar ini adalah Rock')
    elif classes[0][2] == 1:
        print('Gambar ini adalah Scissors')
    else:
        print('tidak diketahui')
```

Hasil Pengujian

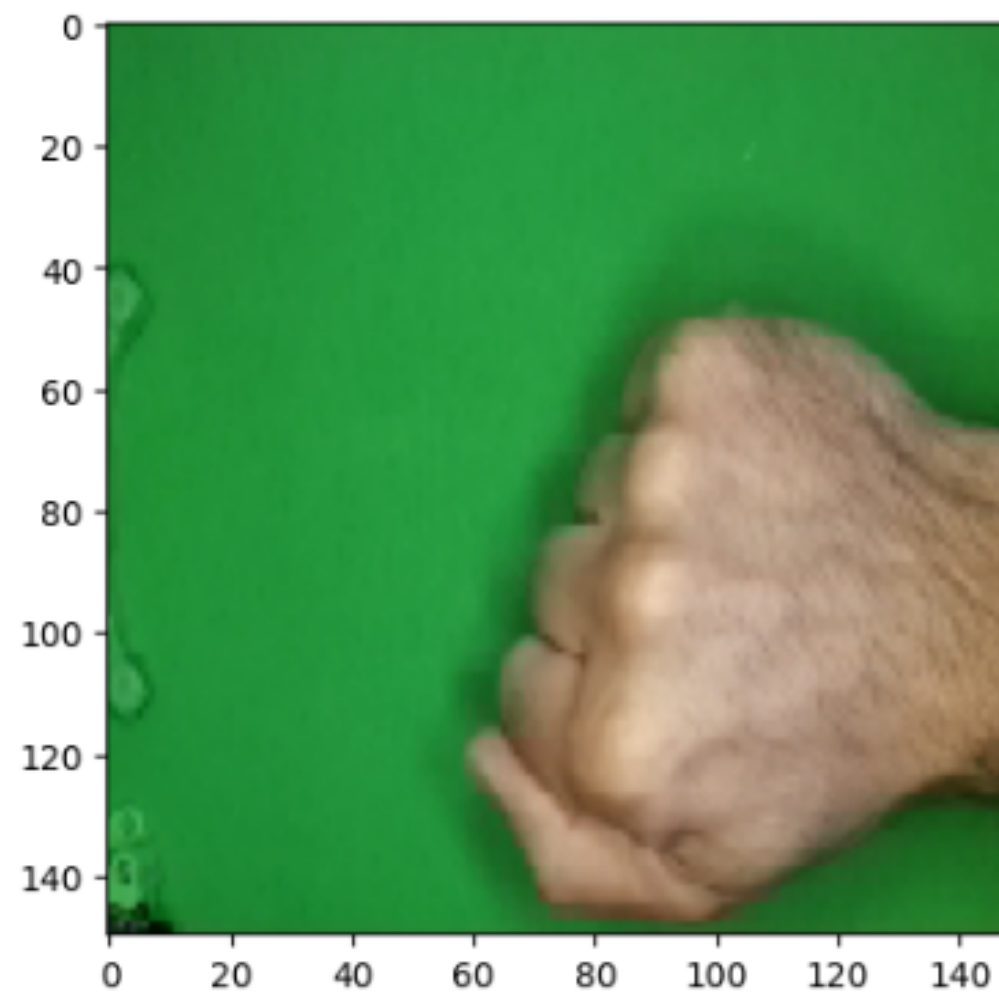
Prediksi Kertas

Gambar ini adalah Paper



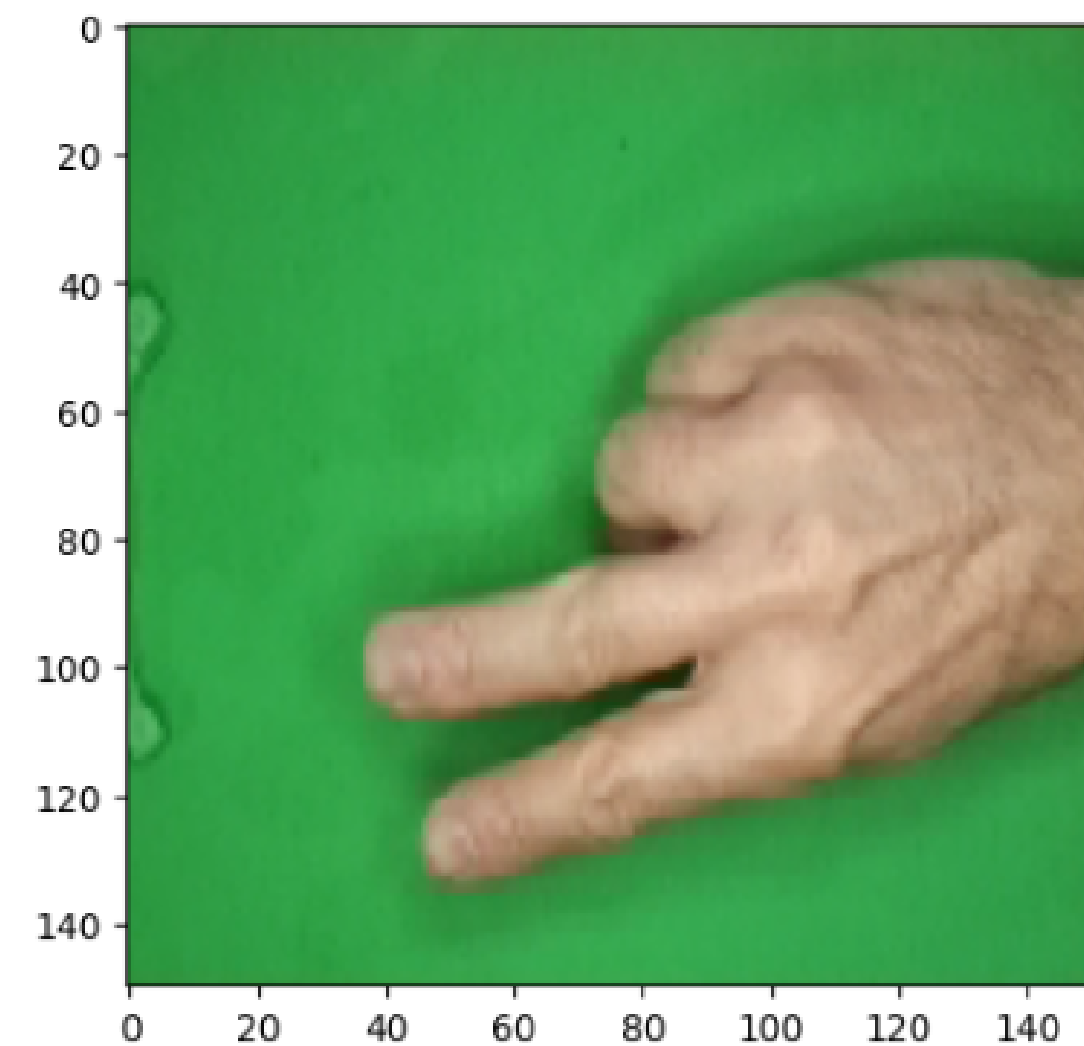
Prediksi Batu

00nKV8oHuTG120gq.png
Gambar ini adalah Rock



Prediksi Gunting

00nKV8oHuTG120gq.png
Gambar ini adalah Scissors

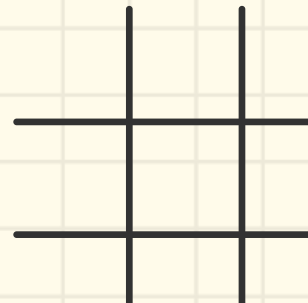
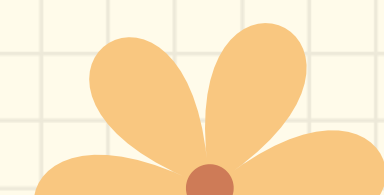


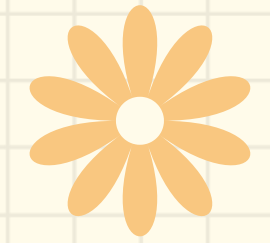
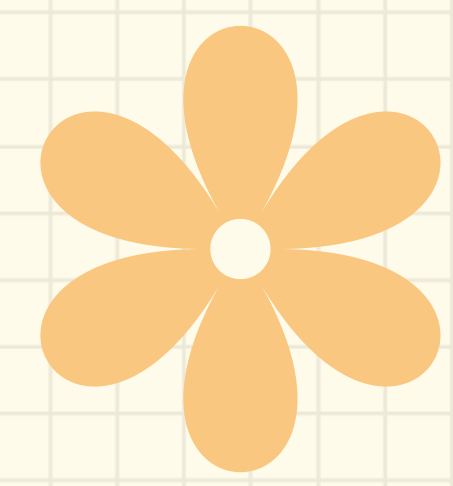


Penjelasan

Pada proyek ini, menggunakan library utama adalah Tesorflow. Daset yang digunakan berjumlah 2188 yang terdiri atas gambar tangan membentuk gunting, batu dan kertas. Dataset dibagi menjadi 60% data latih dan 40% data validasi.

Model sequential yang dihasilkan mencapai tingkat akurasi 97% yang berarti model menunjukkan bekerja cukup baik untuk melakukan prediksi.





Thanks You

Selviana

selvianazainuddin1010@gmail.com

