Artwork/Project Title

Batik Classification Using Convolutional Neural Network

Year Accomplished

2023

Role/Position

Data Scientist

Publication Link

https://drive.google.com/drive/folders/1Cpe

Artwork/Project Description

This project was part of my Deep Learning course's mid-semester exam. The objective was to implement classification using a Convolutional Neural Network (CNN) on batik data. To begin, I conducted data exploration, data cleaning, data preparation, and developed a baseline model. After that, I performed architecture modifications and parameter tuning to achieve the best results. The main challenge was the limited amount of data available and the low quality of the images. To address this, I diligently investigated the root causes and explained why the obtained results were not significantly high. At the end, my hard work paid off, and I received a perfect score of 100 from my instructor.

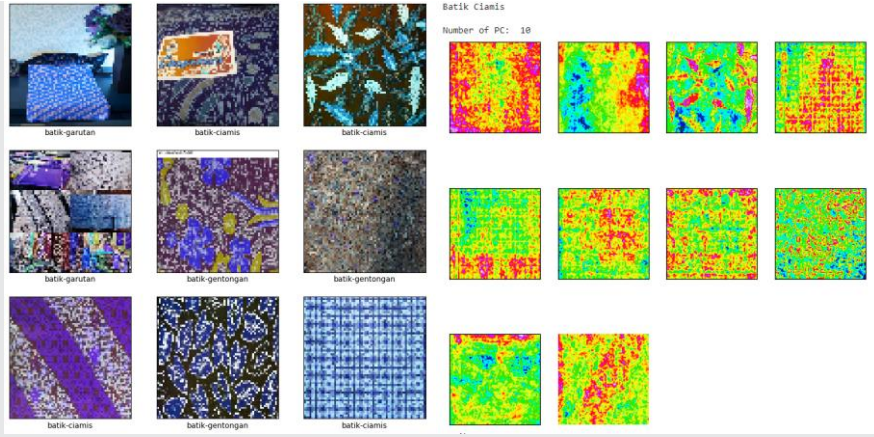

Figure 1 Data Preview

Figure 2 Eugen images

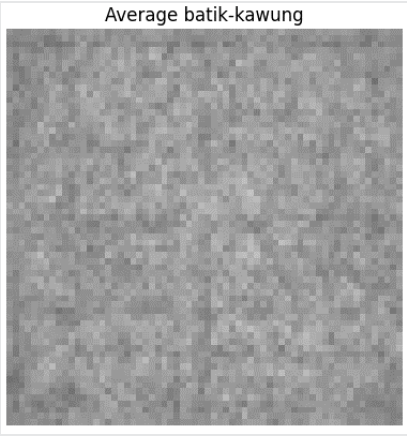

Figure 3 Average Image

```
model = keras.Sequential()
model.add(layers.Conv2D(filters=64, kernel_size=(5, 5),
                        strides=(1, 1), activation="relu", padding="valid",
                        input_shape=(64, 64, 3)))
model.add(layers.MaxPool2D(pool_size=(13, 13), strides= (2, 2), padding="valid"))

model.add(layers.Conv2D(filters=256, kernel_size=(5, 5),
                        strides=(1, 1), activation="relu",
                        padding="same"))

model.add(layers.MaxPool2D(pool_size=(2, 2), strides=(2, 2)))

model.add(layers.Conv2D(filters=384, kernel_size=(3, 3),
                        strides=(1, 1), activation="relu",
                        padding="same"))

model.add(layers.Conv2D(filters=384, kernel_size=(3, 3),
                        strides=(1, 1), activation="relu",
                        padding="same"))

model.add(layers.Conv2D(filters=192, kernel_size=(3, 3),
                        strides=(1, 1), activation="relu",
                        padding="same"))

model.add(layers.MaxPool2D(pool_size=(1, 1), strides=(1, 1)))
model.add(layers.Flatten())
model.add(layers.Dense(4096, activation="relu"))

model.add(layers.Dense(4096, activation="relu"))

model.add(layers.Dense(5, activation="softmax"))

optimizer = keras.optimizers.Adam(learning_rate=0.1)
model.compile(loss='categorical_crossentropy',
              optimizer=optimizer,
              metrics=['accuracy'])
model.summary()
```

Figure 4 Convolutional Neural Network

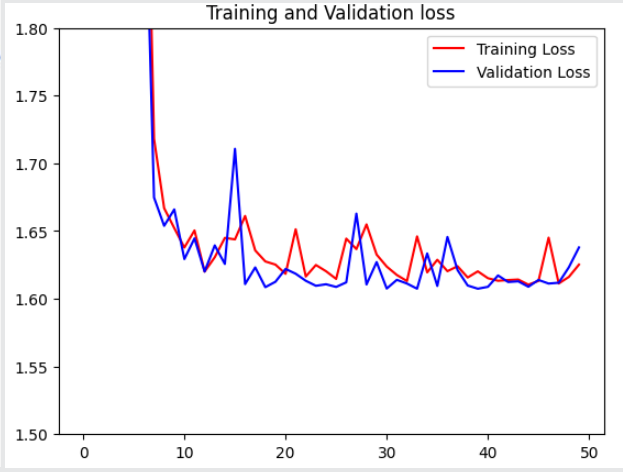

Figure 5 Training vs Validation Loss

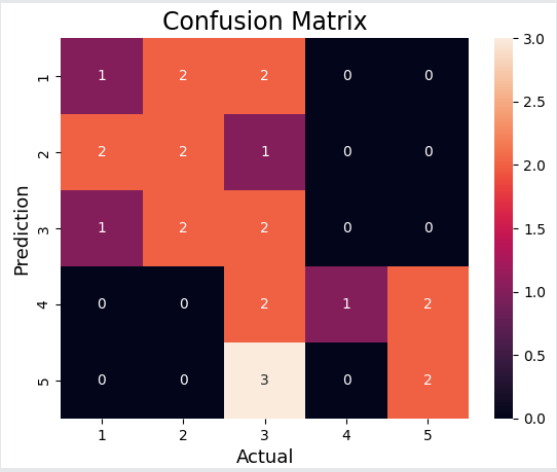

Figure 6 Confusion Matrix

Belinda Mutiara

BINUS Students 5th Semester

belinda.mutiara@binus.ac.id

phone: 0812 8308 7870
line: belinda.mutiara
linkedin: www.linkedin.com/in/belinda-mutiara

Portfolio Submission for

BINUS Internship Track 2024