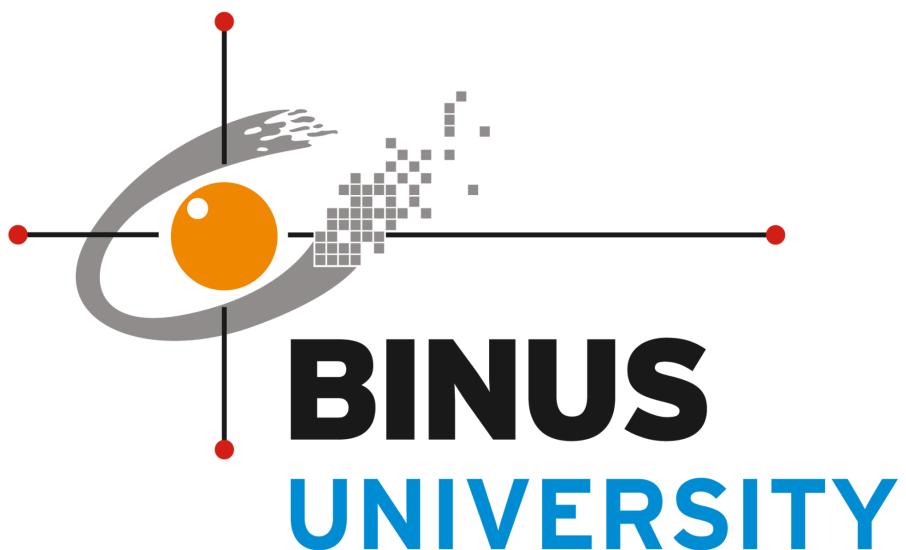


Batik Pattern Classification



Final Project Deep Learning

Anggota Kelompok :

- Vincent - 2702279325
- Vincent Antonio Ekachandra - 2702240891
- Evan Fanuel - 2702327116

I. Latar Belakang

Batik merupakan warisan budaya Indonesia yang memiliki nilai seni dan filosofi tinggi, dengan beragam motif yang mencerminkan identitas daerah, sejarah, serta makna simbolik tertentu. Seiring berkembangnya industri kreatif dan digitalisasi, kebutuhan akan sistem otomatis untuk mengenali dan mengklasifikasikan motif batik semakin meningkat. Proses identifikasi batik secara manual membutuhkan keahlian khusus dan waktu yang tidak sedikit, sehingga berpotensi menimbulkan kesalahan subjektif, terutama ketika jumlah data dan variasi motif semakin besar.

Perkembangan teknologi *deep learning*, khususnya *Convolutional Neural Network* (CNN), telah menunjukkan performa yang sangat baik dalam pengolahan citra digital. Salah satu arsitektur CNN yang populer adalah **ResNet50**, yang mampu mengatasi permasalahan *vanishing gradient* melalui penggunaan *residual connections*. Dengan memanfaatkan teknik **transfer learning** dan **fine-tuning**, model ResNet50 yang telah dilatih pada dataset berskala besar seperti ImageNet dapat diadaptasi untuk klasifikasi motif batik dengan jumlah data yang relatif terbatas, namun tetap menghasilkan akurasi yang tinggi.

Agar hasil penelitian dapat dimanfaatkan secara praktis, diperlukan sebuah aplikasi yang mudah digunakan oleh pengguna umum. **Streamlit** dipilih sebagai media *deployment* karena mampu menyajikan antarmuka web yang interaktif, ringan, dan cepat dalam mengintegrasikan model *machine learning*. Dengan adanya aplikasi berbasis Streamlit, pengguna dapat mengunggah citra batik dan memperoleh hasil klasifikasi secara langsung. Diharapkan sistem ini dapat membantu pelestarian budaya batik, mendukung edukasi masyarakat, serta menjadi solusi inovatif dalam pengenalan motif batik berbasis teknologi kecerdasan buatan.

II. Dataset

Dataset yang digunakan dalam proyek ini merupakan dataset citra batik yang telah terstruktur dengan baik ke dalam dua subset utama, yaitu **data latih (train)** dan **data uji (test)**. Setiap subset dibagi lagi ke dalam beberapa folder kelas yang merepresentasikan motif batik tertentu, misalnya *Aceh_Pintu_Aceh* dan motif lainnya. Struktur folder ini mengikuti standar input *image classification* pada Keras, sehingga memudahkan proses pemuatan data secara otomatis. Pembagian dataset ke dalam train dan test menunjukkan bahwa proyek ini telah memperhatikan prinsip evaluasi model yang baik, yaitu menguji performa model pada data yang tidak pernah dilihat sebelumnya. Keberadaan citra batik dalam berbagai variasi visual juga membantu model dalam mempelajari karakteristik motif secara lebih general.

Preview Dataset :



Dataset Handling :

```
IMG_SIZE = (224, 224)
BATCH_SIZE = 16
EPOCHS = 50
NUM_CLASSES = 20

TRAIN_DIR = "./batik_dataset/train/"
TEST_DIR = "./batik_dataset/test/"
```

```

train_ds = image_dataset_from_directory(
    TRAIN_DIR,
    image_size=IMG_SIZE,
    batch_size=BATCH_SIZE,
    label_mode="categorical",
    shuffle=True
)

test_ds = image_dataset_from_directory(
    TEST_DIR,
    image_size=IMG_SIZE,
    batch_size=BATCH_SIZE,
    label_mode="categorical",
    shuffle=False
)

class_names = train_ds.class_names
print("Classes:", class_names)

```

```

AUTOTUNE = tf.data.AUTOTUNE
train_ds = train_ds.prefetch(buffer_size=AUTOTUNE)
test_ds = test_ds.prefetch(buffer_size=AUTOTUNE)

```

Preprocessing :

```

data_augmentation = tf.keras.Sequential([
    layers.RandomFlip("horizontal"),
    layers.RandomRotation(0.1),
    layers.RandomZoom(0.1),
    layers.RandomContrast(0.1),
])

```

III. Model

Model yang digunakan dalam proyek ini adalah ResNet50 dengan pendekatan transfer learning dan fine-tuning, yang disimpan dalam file batik_classification_model.keras. ResNet50 dipilih karena kemampuannya dalam mengekstraksi fitur visual yang kompleks melalui arsitektur deep residual network. Dengan memanfaatkan bobot awal dari dataset ImageNet, model dapat mengenali pola dasar seperti tekstur, bentuk, dan warna batik secara lebih efektif meskipun jumlah dataset terbatas. Proses fine-tuning dilakukan dengan membuka beberapa layer terakhir untuk dilatih ulang agar model lebih adaptif terhadap karakteristik khusus motif batik. Informasi kelas disimpan dalam file class_names.npy, yang berfungsi sebagai pemetaan antara output model dan label kelas.

ResNet50 Modelling :

```

base_model = ResNet50(
    weights="imagenet",
    include_top=False,
    input_shape=(224, 224, 3)
)

base_model.trainable = False # Freeze backbone

inputs = layers.Input(shape=(224, 224, 3))
x = data_augmentation(inputs)
x = tf.keras.applications.resnet50.preprocess_input(x)

x = base_model(x, training=False)
x = layers.GlobalAveragePooling2D()(x)
x = layers.BatchNormalization()(x)
x = layers.Dense(256, activation="relu")(x)
x = layers.Dropout(0.5)(x)

outputs = layers.Dense(NUM_CLASSES, activation="softmax")(x)

model = models.Model(inputs, outputs)

model.compile(
    optimizer=Adam(learning_rate=1e-4),
    loss="categorical_crossentropy",
    metrics=["accuracy"]
)

model.summary()

```

Model Architecture :

Layer (type)	Output Shape	Param #
Connected to		
input_layer_1 (InputLayer)	(None, 224, 224, 3)	0
sequential input_layer_1[0]... (Sequential)	(None, 224, 224, 3)	0

get_item (GetItem)	(None, 224, 224)	0
sequential[0][0]		
get_item_1	(None, 224, 224)	0
sequential[0][0]		
(GetItem)		
get_item_2	(None, 224, 224)	0
sequential[0][0]		
(GetItem)		
stack (Stack)	(None, 224, 224, 3)	0
get_item[0][0],		
get_item_1[0][0],		
get_item_2[0][0]		
add (Add)	(None, 224, 224, 3)	0
stack[0][0]		
resnet50	(None, 7, 7, 23,587,712)	23,587,712
add[0][0]		
(Functional)	2048)	
global_average_poo...	(None, 2048)	0
resnet50[0][0]		
(GlobalAveragePool...)		
batch_normalization	(None, 2048)	8,192
global_average_p...		
(BatchNormalizatio...)		
dense (Dense)	(None, 256)	524,544
batch_normalizat...		

dropout (Dropout)	(None, 256)		0
dense[0][0]			
dense_1 (Dense)	(None, 20)		5,140
dropout[0][0]			

Model Training :

```

early_stopping = tf.keras.callbacks.EarlyStopping(
    monitor="val_loss",
    patience=5,
    restore_best_weights=True
)

history = model.fit(
    train_ds,
    validation_data=test_ds,
    epochs=EPOCHS,
    callbacks=[early_stopping]
)

```

Fine Tuned-Training :

```

# Unfreeze model
base_model.trainable = True

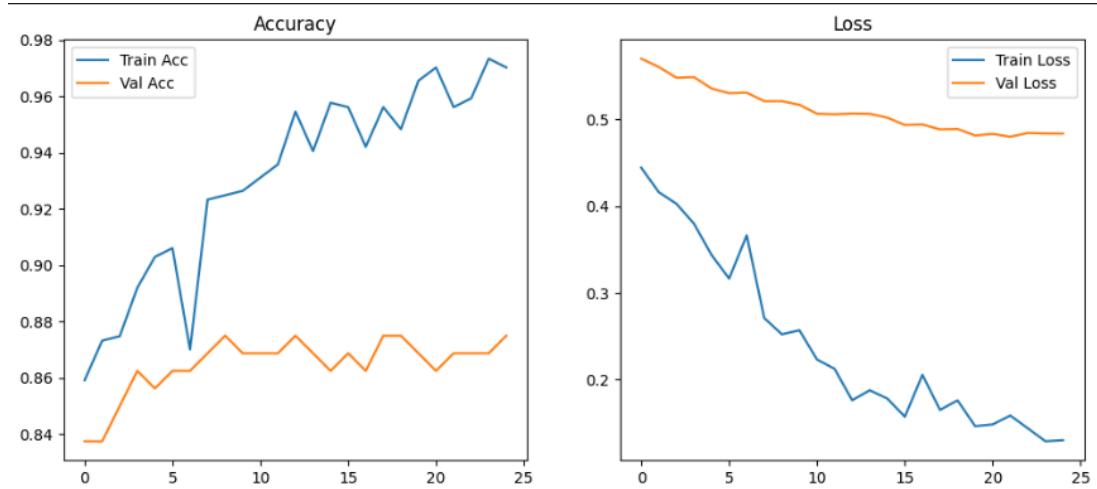
# Freeze sebagian layer (agar tidak overfitting)
# unfreeze 30 layer terakhir
for layer in base_model.layers[:-30]:
    layer.trainable = False

model.compile(
    optimizer=tf.keras.optimizers.Adam(learning_rate=1e-5),
    loss="categorical_crossentropy",
    metrics=["accuracy"]
)

fine_tune_history = model.fit(
    train_ds,
    validation_data=test_ds,
    epochs=25,
    callbacks=[early_stopping]
)

```

Graphs :



IV. Evaluation/Test

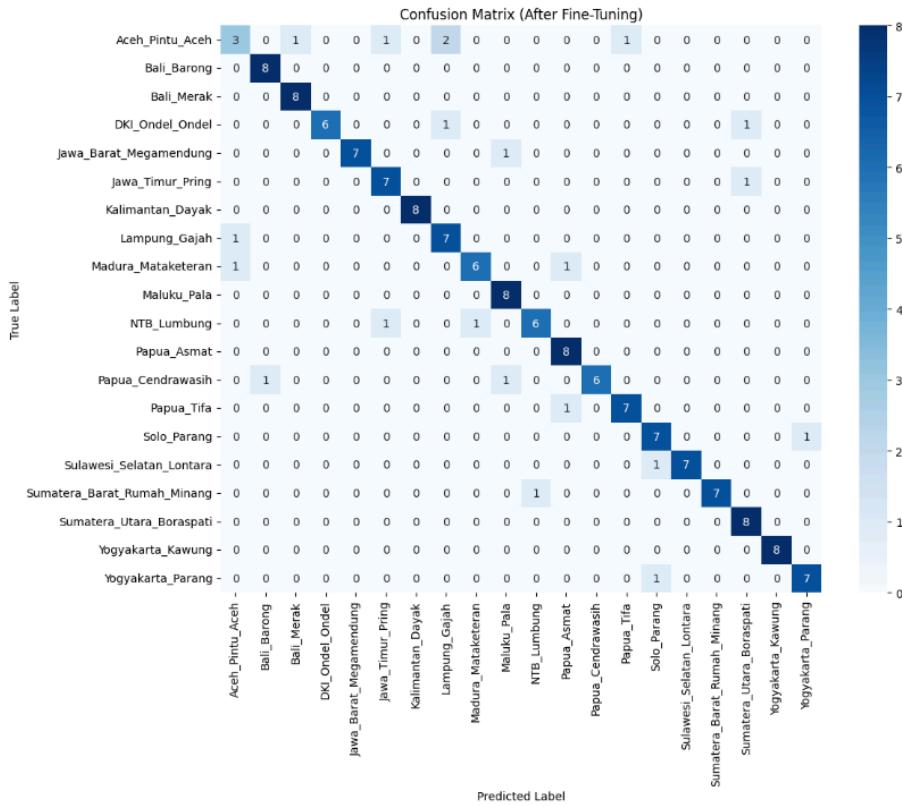
Evaluasi model dilakukan dengan menggunakan data uji (test) yang terpisah dari data latih, sehingga hasil pengujian mencerminkan kemampuan generalisasi model. Proses evaluasi dan eksperimen model dapat dilihat pada file ProjectDL.ipynb, yang berisi tahapan pelatihan, validasi, serta pengujian performa model. Metode ini memungkinkan analisis terhadap akurasi dan kesalahan klasifikasi pada masing-masing motif batik. Dengan adanya pemisahan data train dan test yang jelas, proyek ini telah menerapkan praktik machine learning yang baik untuk menghindari overfitting dan memastikan reliabilitas hasil.

Model Evaluation :

```
final_test_loss, final_test_acc = model.evaluate(test_ds)
print(f"Final Test Accuracy : {final_test_acc:.4f}")

10/10 ━━━━━━━━ 15s 1s/step - accuracy: 0.8687 - loss: 0.4800
Final Test Accuracy : 0.8687
```

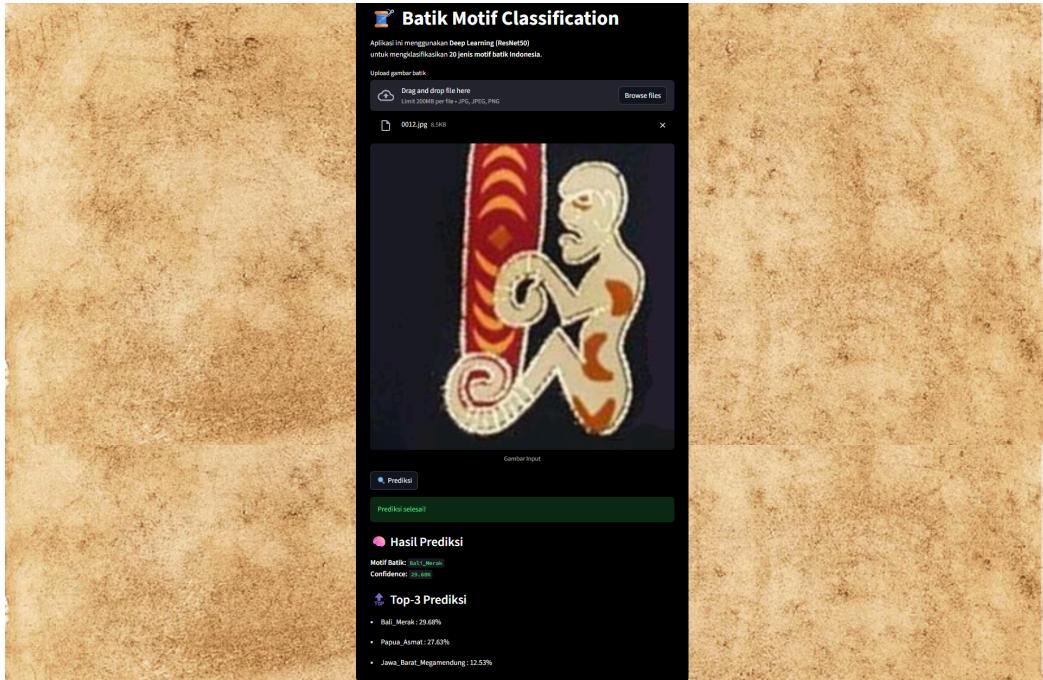
Confusion Matrix :



Classification Report :

	precision	recall	f1-score	support
Aceh_Pintu_Aceh	0.60	0.38	0.46	8
Bali_Barong	0.89	1.00	0.94	8
Bali_Merak	0.89	1.00	0.94	8
DKI_Ondel_Ondel	1.00	0.75	0.86	8
Jawa_Barat_Megamendung	1.00	0.88	0.93	8
Jawa_Timur_Pring	0.78	0.88	0.82	8
Kalimantan_Dayak	1.00	1.00	1.00	8
Lampung_Gajah	0.70	0.88	0.78	8
Madura_Mataketeran	0.86	0.75	0.80	8
Maluku_Pala	0.80	1.00	0.89	8
NTB_Lumbung	0.86	0.75	0.80	8
Papua_Asmat	0.80	1.00	0.89	8
Papua_Cendrawasih	1.00	0.75	0.86	8
Papua_Tifa	0.88	0.88	0.88	8
Solo_Parang	0.78	0.88	0.82	8
Sulawesi_Selatan_Lontara	1.00	0.88	0.93	8
Sumatera_Barat_Rumah_Minang	1.00	0.88	0.93	8
Sumatera_Utara_Boraspatti	0.80	1.00	0.89	8
Yogyakarta_Kawung	1.00	1.00	1.00	8
Yogyakarta_Parang	0.88	0.88	0.88	8
accuracy			0.87	160
macro avg		0.87	0.87	160
weighted avg		0.87	0.87	160

V. Deployment



Link Deployment : <https://website-batik-classification.streamlit.app/>

Link Github: <https://github.com/aidennz/Batik-Classification.git>

VI. Reflection

Secara keseluruhan, proyek ini telah mengintegrasikan seluruh tahapan *deep learning pipeline* dengan baik, mulai dari pengelolaan dataset, pelatihan model, evaluasi, hingga deployment. Penggunaan ResNet50 dengan fine-tuning menunjukkan pemahaman yang baik terhadap pemanfaatan transfer learning untuk permasalahan klasifikasi citra. Selain itu, deployment menggunakan Streamlit memberikan nilai tambah karena membuat sistem lebih aplikatif dan mudah diakses. Ke depan, proyek ini masih dapat dikembangkan lebih lanjut, misalnya dengan menambah jumlah dan variasi data batik, meningkatkan evaluasi menggunakan metrik lain seperti confusion matrix, atau mengoptimalkan tampilan dan fitur aplikasi agar lebih informatif bagi pengguna.

VII. Reference

- <https://www.kaggle.com/datasets/hendryhb/batik-nusantara-batik-indonesia-dataset>
- Meranggi, D. G. T., Yudistira, N., & Sari, Y. A. (2021). *Batik classification using convolutional neural network with data improvements*. *Journal of Open Innovation: Technology, Market, and Complexity*. <https://joiv.org/index.php/joiv/article/view/716/402>
- Rangkuti, A. H., Harjoko, A., & Putra, A. (2021). *A novel reliable approach for image batik classification that invariant with scale and rotation using MU2ECS-LBP algorithm*. *Procedia Computer Science*, 179, 214–221. <https://www.sciencedirect.com/science/article/pii/S1877050921001058>