

Klasifikasi Batik Menggunakan Algoritma CNN (*Convolutional Neural Network*)

Dias Nur Ramadhan¹, Rio Aldi Erwanto², Rony Tan Enwan³

^{1,2,3}Teknik Informatika, Fakultas Teknik dan Ilmu Komputer, Universitas Nusantara PGRI Kediri

E-mail: ¹[*¹diasnurramadhan@gmail.com](mailto:diasnurramadhan@gmail.com), ²rioaldierwanto@gmail.com, ³ronytanwan@gmail.com

Abstrak – Batik adalah warisan budaya Indonesia, yang telah menjadi ciri khas orang Indonesia. Banyak corak dan motif batik, dan setiap corak atau motif memiliki karakteristik unik. Sebagai bagian dari upaya untuk melestarikan batik, penelitian tentang klasifikasi batik digunakan untuk mengidentifikasi motif citra batik. Dengan demikian, penelitian ini dilakukan menggunakan metode pembelajaran dengan data citra batik. Salah satu metode klasifikasi yang akurat dan efisien dengan tingkat akurasi tinggi adalah Convolutional Neural Network (CNN). Convolutional Neural Network (CNN) adalah kombinasi teknik deep learning dan jaringan syaraf tiruan. CNN terdiri dari tiga lapisan utama: lapisan convolutional, lapisan pooling, dan lapisan penuh terhubung. Setelah melakukan pembahasan, analisis ini diakhiri dengan kesimpulan mengenai algoritma Convolutional Neural Network dalam mengklasifikasikan Batik. Berdasarkan penelitian-penelitian sebelumnya, dapat diketahui bahwa convolution neural network dapat bekerja dengan baik untuk klasifikasi citra dengan dataset yang besar. Dengan mengevaluasi metode yang telah dijelaskan dengan mempertimbangkan arsitektur dan tingkat akurasi maka klasifikasi batik menunjukkan bahwa ukuran citra, kualitas citra, pola citra mempengaruhi proses klasifikasi batik. Hasil dari penelitian ini mendapatkan akurasi model pada data pengujian tercapai sebesar 0.5263 atau sekitar 52.63%.

Kata Kunci —batik, klasifikasi, convolutional neural network, image, deep learning.

1. PENDAHULUAN

Batik merupakan salah satu ikon budaya dari Indonesia yang telah menerima penghargaan sebagai warisan budaya dari UNESCO. Batik, dalam bidang seni rupa, dianggap sebagai lukisan dua dimensi di mana kain digunakan sebagai media lukis. Batik memiliki nilai dan kharisma yang tinggi. Berbagai motif batik menggambarkan animisme dan dinamisme, dan telah diwariskan dari generasi ke generasi [1]. Batik bagi Kamus Besar Bahasa Indonesia merupakan kain bergambar yang pembuatannya secara spesial dengan menuliskan ataupun menerangkan malam pada kain itu, setelah itu pembuatannya secara spesial dengan menuliskan ataupun menerakan malam pada kain itu, setelah itu pengolahannya lewat proses tertentu [2]. Setiap daerah di Indonesia memiliki batik dengan motif yang berbeda, dan setiap motif memiliki filosofi dan makna tersendiri. Ciri khas dan makna yang ingin disampaikan membuat setiap motif unik [3]. Oleh karena itu batik memang sangat istimewa, corak dan motifnya tidak terlepas dari daerah asal mereka.

Untuk menjaga salah satu warisan budaya Indonesia ini, diperlukan penelitian yang berkesinambungan. Oleh karena itu, diperlukan suatu teknik untuk mengeluarkan fitur batik berdasarkan motifnya. Salah satu bidang penelitian yang mendapat banyak pengembangan di ilmu komputer adalah klasifikasi batik. Klasifikasi batik adalah salah satu topik penelitian yang paling banyak dikembangkan dalam bidang ilmu komputer. Penelitian ini juga relevan dengan topik yang masih aktual, yaitu ekstraksi fitur, yang digunakan untuk mewakili karakteristik suatu gambar batik sehingga dapat mengidentifikasi gambar dengan benar. Meskipun demikian, kain batik memiliki beragam motif, sehingga sulit untuk menemukan gambar batik yang sesuai dengan variasi motifnya. Oleh karena itu, dibutuhkan suatu teknik untuk mengidentifikasi karakteristik batik berdasarkan motifnya. [4].

Pengenalan pola adalah suatu proses untuk mengambil dan mengklasifikasikan data, di mana data tersebut dapat berupa gambar, tulisan, suara, angka, dan lain-lain. Pengenalan pola merupakan salah satu bidang yang fokus pada metode pengklasifikasian objek ke dalam kelas-kelas tertentu untuk menyelesaikan masalah tertentu. Pengenalan pola juga melibatkan beberapa proses, seperti pengambilan citra, preprocessing yang digunakan untuk menghilangkan noise, ekstraksi fitur, dan klasifikasi. Dengan demikian, pengenalan pola banyak digunakan sebagai alat untuk mengklasifikasikan data[5].

Convolutional Neural Network adalah algoritma yang telah menjadi populer sejak klasifikasi ImageNet 2012 kompetisi benchmark. Hingga saat ini, CNN menjadi pilihan untuk metode yang menggunakan data gambar sebagai input karena kinerjanya yang kinerja yang baik. Beberapa pengenalan yang berkaitan dengan pengenalan tulisan telah dilakukan dengan rata-rata akurasi hasil yang diperoleh di atas 80% [6]. *Convolutional Neural Network (CNN)* adalah salah satu algoritma canggih yang dimiliki oleh jaringan syaraf dan memiliki kelas model yang baik untuk mengenali teks tulisan tangan. Jaringan syaraf bekerja seperti otak manusia, yang dapat dilatih untuk meningkatkan pengetahuan untuk mendapatkan akurasi yang tinggi. Dengan menganalisa setiap piksel gambar dan mencocokkannya dengan data yang ada, metode ini cocok untuk dokumen dan teks

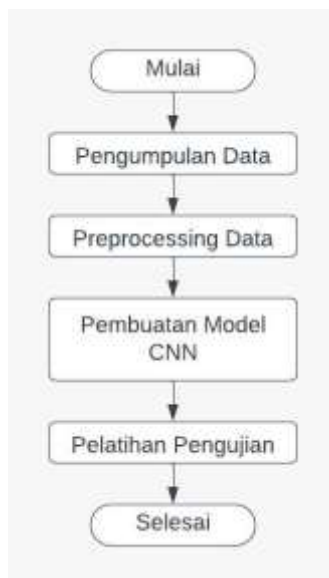
yang rusak. Jaringan syaraf sangat ideal untuk masalah-masalah spesifik seperti data pasar saham atau menemukan Sejauh ini *neural network* merupakan metode yang paling efisien dibandingkan dengan metode lainnya [7].

Pengetahuan mengenai identifikasi jenis motif batik sebagian besar dimiliki oleh kelompok tertentu, hal ini disebabkan oleh keragaman motif batik yang hampir serupa di setiap daerahnya. Mengatasi masalah ini, salah satu pendekatan yang digunakan adalah melalui pengenalan pola. Metode ini sangat efektif diterapkan dalam klasifikasi motif batik [8].

Berdasarkan permasalahan tersebut, maka dibuatlah penelitian ini dengan judul ” Klasifikasi Batik Menggunakan Algoritma *Convolutional Neural Network* “. Klasifikasi motif batik dilakukan dengan memanfaatkan algoritma *Convolutional Neural Network* (CNN) sebagai teknik pengenalan pola, terutama dalam klasifikasi citra batik. CNN adalah suatu model *deep learning* yang mampu melakukan pembelajaran otomatis untuk mengenali objek, mengekstraksi fitur objek, dan melakukan klasifikasi, serta dapat diterapkan pada citra beresolusi tinggi dengan distribusi nonparametrik. Penerapan metode CNN ini bertujuan untuk mengidentifikasi pola batik. Proses klasifikasi menggunakan Algoritma *Convolutional Neural Network*, dan metodologi yang digunakan didasarkan pada tinjauan literatur, dengan merinci hasil kajian dari beberapa jurnal yang membahas Algoritma *Convolutional Neural Network* dalam konteks klasifikasi, dan memberikan simpulan mengenai utilitas algoritma tersebut.

2. METODE PENELITIAN

Metode penelitian yang digunakan adalah metode *Waterfall*. metode *waterfall* dilakukan dengan menggunakan pendekatan sistematis, yang dimulai dengan tahap pengumpulan data, pre-processing, penerapan model dan pengujian yang dilakukan secara berurutan.



Gambar 1. Diagram Alir

2.1 Pengumpulan Data

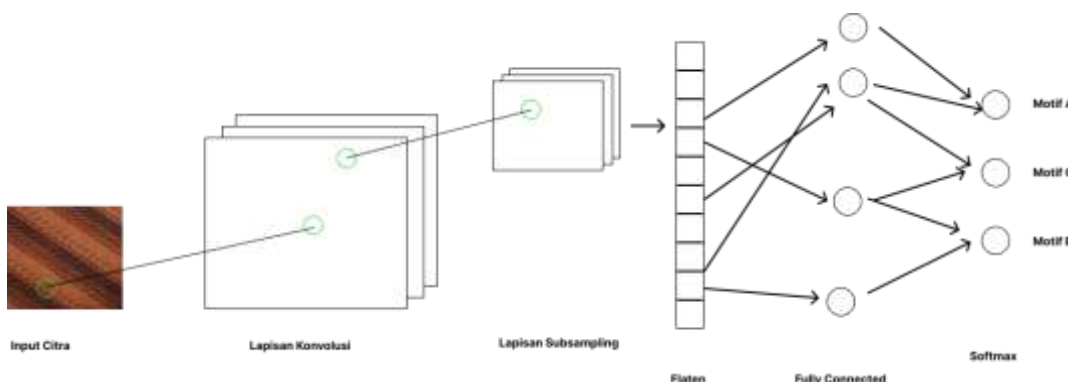
Dataset yang digunakan dalam penelitian ini diambil dari Kaggle. Dataset ini terdiri dari gambar citra batik yang mencakup berbagai jenis motif batik dari berbagai daerah di Indonesia. Citra dari dataset tersebut akan disimpan kedalam folder sesuai dengan jenis motif masing-masing batik.

2.2 *Preprocessing Data*

Data yang telah dikumpulkan akan menjalani proses *preprocessing* sebelum digunakan dalam CNN. Penelitian ini melibatkan dua tahap *preprocessing*, yaitu penyesuaian ukuran piksel dan proses augmentasi. Penyesuaian ukuran piksel diperlukan untuk memastikan bahwa setiap citra memiliki ukuran piksel yang seragam, karena CNN membutuhkan ukuran piksel yang konsisten pada setiap citra, sementara citra motif batik yang telah diunduh memiliki variasi ukuran piksel. Proses augmentasi pada citra motif batik diperlukan untuk

memperkaya dataset dengan variasi yang lebih banyak. Hal ini dilakukan karena CNN memiliki tingkat akurasi yang tinggi saat menggunakan dataset dengan jumlah yang besar [9].

2.3 Pembuatan Model

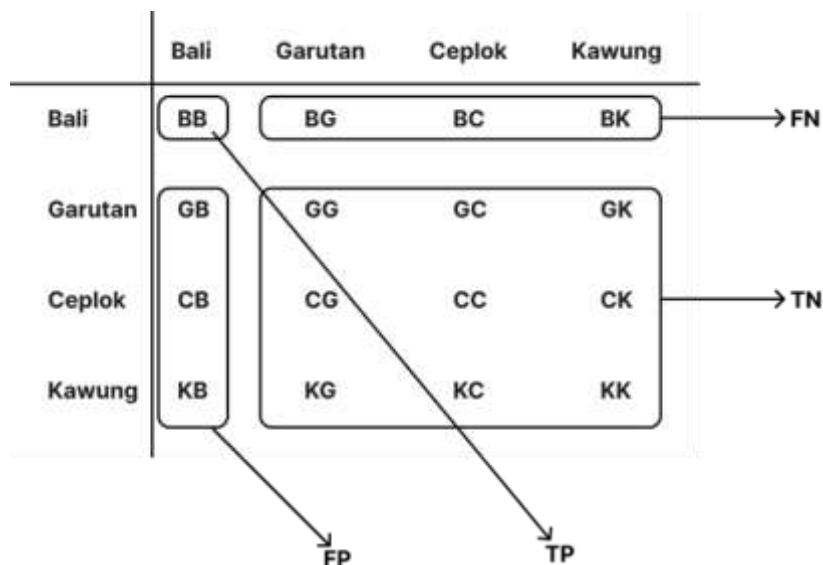


Gambar 2. Arsitektur CNN

Secara umum, model CNN (Gambar 2) terdiri dari dua proses utama, yaitu ekstraksi ciri dan klasifikasi. Proses ekstraksi ciri melibatkan lapisan konvolusi yang bertugas mencari fitur atau karakteristik menggunakan filter, serta lapisan sub-sampling yang berperan mengurangi dimensi dari fitur tanpa menghilangkan informasi penting. Sementara itu, proses klasifikasi melibatkan lapisan flatten yang bertugas mengubah dimensi dari fitur yang berbentuk matriks menjadi vektor dengan satu dimensi. *Fully connected layer* berfungsi untuk mentransformasikan data dari dimensi yang ada agar dapat diolah secara linear sehingga dapat dilakukan klasifikasi. Sementara itu, softmax berfungsi untuk mengklasifikasikan kelas yang memiliki lebih dari dua kelas dalam suatu masalah klasifikasi [10].

Dalam penelitian ini, model CNN yang digunakan untuk mengklasifikasikan citra motif batik dibuat dalam bentuk model sequential menggunakan library Keras. Lebih dari satu model CNN dibangun dengan variasi pada ekstraksi ciri. Beberapa skenario model CNN dikembangkan berdasarkan jumlah lapisan konvolusi, ukuran kernel, penggunaan padding, stride, serta jenis sub-sampling.

Pada Penelitian ini penerapan model CNN akan diuji menggunakan *Confusion Matrix*. *Confusion matrix* adalah tabel atau matriks yang menunjukkan hasil dari proses klasifikasi yang membandingkan antara hasil klasifikasi aktual dan hasil prediksi. Matriks ini memiliki ukuran sesuai dengan jumlah kelas yang ada dan berguna untuk melakukan perhitungan evaluasi klasifikasi, seperti akurasi, dalam konteks data mining [11] [12]. Contoh pada gambar 3, Confusion Matrix dengan data sample yang memiliki 4 kelas yaitu bali, garutan, ceplik, dan kawung.



Gambar 3. Confusion Matrix

Dimana:

FP adalah false positif yaitu jumlah data positif yang terklasifikasi dengan benar oleh system

TP adalah true positif yaitu jumlah data negatif yang terklasifikasi dengan benar oleh system.

FN adalah false negative yaitu jumlah data negative namun terklasifikasi salah oleh system

TN adalah true negative yaitu jumlah data positif namun terklasifikasi salah oleh system

Confusion matrix digunakan untuk mengetahui nilai akurasi, precision, dan recall dari model klasifikasi motif batik.

$$Akurasi = \frac{TP\ Total}{Total\ Dataset} \dots\dots\dots(1)$$

$$Precision = \frac{TP}{TP+FP} = \frac{TP}{Total\ prediksi} \dots\dots\dots(2)$$

$$Recall = \frac{TP}{TP+FN} = \frac{TP}{Total\ aktual} \dots\dots\dots(3)$$

Dalam *Confusion matrix ini*, TP mencakup jumlah prediksi yang tepat sesuai dengan kelasnya sendiri. Di sisi lain, TN mencakup jumlah data yang benar-benar bukan bagian dari kelas tersebut. FN mewakili prediksi yang salah mengenai kelasnya sendiri. FP adalah jumlah prediksi yang keliru mengenai kelasnya sendiri.

2.4 Pengujian

Tahap pengujian merupakan tahap evaluasi program yang telah dibuat dengan memanfaatkan citra yang telah disiapkan sebelumnya, pada tahap ini digunakan data image baru untuk menentukan nilai predict data tersebut berdasarkan proses perhitungan dengan metode CNN didalam program yang telah dibuat.

3. HASIL DAN PEMBAHASAN

3.1 Import Library

Pada penelitian ini, kami mengimpor berbagai *library* dan modul yang diperlukan untuk menjalankan model jaringan saraf tiruan, termasuk *TensorFlow*, *Keras*, dan modul-modul lain yang terkait dengan pemrosesan gambar, pengolahan data, dan visualisasi.

```
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import backend as K
from tensorflow.keras.layers import Dense, Activation, Dropout, Conv2D, MaxPooling2D, BatchNormalization, Flatten, Input
from tensorflow.keras.optimizers import Adam, Adamax
from tensorflow.keras.metrics import categorical_crossentropy
from tensorflow.keras import regularizers
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.models import Model, load_model, Sequential
from keras.callbacks import ModelCheckpoint
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
from matplotlib.pyplot import imshow
import os
import seaborn as sns
sns.set_style('darkgrid')
from sklearn.metrics import confusion_matrix, classification_report
```

Gambar 4. Import Library

3.2 Pengambilan Dataset

Pada penelitian ini, kami menggunakan dataset dari Kaggle, setelah kami mengunduh dataset tersebut kami menyimpannya ke dalam google drive, agar data tersebut bisa kami gunakan di dalam google colab.

```
sdir='./drive/MyDrive/dataset/'
filepaths=[]
labels=[]
classlist=os.listdir(sdir)
for klass in classlist:
    classpath=os.path.join(sdir,klass)
    if os.path.isdir(classpath):
        flist=os.listdir(classpath)
        for f in flist:
            fpath=os.path.join(classpath,f)
            filepaths.append(fpath)
            labels.append(klass)
Fseries= pd.Series(filepaths, name='filepaths')
Lseries=pd.Series(labels, name='labels')
df=pd.concat([Fseries, Lseries], axis=1)
print (df.head())
print (df['labels'].value_counts())
```

	filepaths	labels
0	./drive/MyDrive/dataset/batik-bali/1.jpg	batik-bali
1	./drive/MyDrive/dataset/batik-bali/12.jpg	batik-bali
2	./drive/MyDrive/dataset/batik-bali/18.jpg	batik-bali
3	./drive/MyDrive/dataset/batik-bali/11.jpg	batik-bali
4	./drive/MyDrive/dataset/batik-bali/32.jpg	batik-bali

batik-bali 58
batik-garutan 58
batik-ceplok 48
batik-kawung 45
Name: labels, dtype: int64

Gambar 5. Pengambilan Dataset

3.3 Preprocessing Gambar

Tahap ini melibatkan pemrosesan gambar sebelum dilakukan pelatihan model. Gambar-gambar dalam dataset diubah menjadi format yang sesuai untuk dimasukkan ke dalam model jaringan saraf tiruan. Pada tahap ini, dilakukan normalisasi piksel gambar dengan mengubah rentang nilai piksel dari 0-255 menjadi 0-1 menggunakan `rescale=1./255`. Selain itu, gambar-gambar juga dibagi menjadi batch-batch kecil untuk mempercepat proses pelatihan dengan menggunakan generator `ImageDataGenerator`.

```
height=224
width=224
channels=3
batch_size=64

img_shape=(height, width, channels)
img_size=(height, width)
length=len(test_df)
test_batch_size=sorted([int(length/n) for n in range(1,length+1) if length % n ==0 and length/n <=80],reverse=True)[0]
test_steps=int(length/test_batch_size)
print ('test batch size: ',test_batch_size, ' test steps: ', test_steps)

gen=ImageDataGenerator(
    rescale=1./255,
)
train_gen=gen.flow_from_dataframe( train_df, x_col='filepaths', y_col='labels', target_size=img_size, class_mode='categorical',
    color_mode='rgb', shuffle=True, batch_size=batch_size)

validgen=ImageDataGenerator(rescale=1./255)
valid_gen=validgen.flow_from_dataframe( valid_df, x_col='filepaths', y_col='labels', target_size=img_size, class_mode='categorical',
    color_mode='rgb', shuffle=True, batch_size=batch_size)

testgen=ImageDataGenerator(rescale=1./255)
test_gen=testgen.flow_from_dataframe( test_df, x_col='filepaths', y_col='labels', target_size=img_size, class_mode='categorical',
    color_mode='rgb', shuffle=False, batch_size=test_batch_size)
```

Gambar 6. Preprocessing Gambar

3.4 Pembagian Data

Dataset yang digunakan untuk pelatihan model dibagi menjadi tiga bagian: data pelatihan (train), data validasi (validation), dan data pengujian (test). Pada kode di atas, dataset awal dipecah menjadi data pelatihan sebesar 80%, data pengujian sebesar 10%, dan data validasi sebesar 10%.

```
train_split=.8
test_split=.1
dummy_split=test_split/(1-train_split)
train_df, dummy_df=train_test_split(df, train_size=train_split, shuffle=True, random_state=123)
test_df, valid_df=train_test_split(dummy_df, train_size=dummy_split, shuffle=True, random_state=123)
```

Gambar 7. Pembagian Data

3.5 Pelatihan model CNN

Model jaringan saraf tiruan yang digunakan adalah model berbasis Xception pre-trained dengan bobot yang telah dilatih pada dataset ImageNet. Model ini menggunakan lapisan konvolusi, pooling, dropout, dan lapisan-lapisan lain untuk melakukan ekstraksi fitur dan klasifikasi gambar. Pada kode di atas, `base_model` adalah model Xception yang telah dilatih pada dataset ImageNet. Lapisan-lapisan konvolusi, pooling, dan dropout ditambahkan di atas `base_model`, diikuti oleh lapisan Flatten untuk meratakan output menjadi vektor, dan lapisan Dense dengan 15 unit dan fungsi aktivasi softmax untuk melakukan klasifikasi gambar ke dalam 15 kelas yang berbeda. Model ini dikompilasi dengan pengoptimal Adam, loss function `categorical_crossentropy`, dan metrik akurasi.

```
model = tf.keras.Sequential([
    # Note the input shape is the desired size of the image 128x128 with 3 bytes color
    # This is the first convolution
    base_model,
    tf.keras.layers.Conv2D(filters=32, padding='same', kernel_size=3, activation='relu', strides=1),
    tf.keras.layers.MaxPool2D(pool_size=2, strides=2),
    tf.keras.layers.Dropout(rate=0.5),

    tf.keras.layers.Flatten(),
    tf.keras.layers.Dense(4, activation='softmax')
])
model.compile(optimizer=tf.keras.optimizers.Adam(learning_rate=0.001), loss='categorical_crossentropy', metrics='accuracy')
```

Gambar 8. Pelatihan Model CNN

3.6 Pelatihan Model

Model di atas dilatih dengan menggunakan generator `train_gen` dan `valid_gen`. Pelatihan dilakukan selama 100 epoch dengan memantau metrik loss dan akurasi pada data pelatihan dan data validasi.

```
epochs =100
history=model.fit(x=train_gen, epochs=epochs, validation_data=valid_gen)
```

Gambar 9. Pelatihan Model

3.7 Hasil Model

Pada penelian ini, dilakukan perhitungan akurasi menggunakan metode `accuracy_score` untuk model yang telah dilatih dan data pengujian. Hasilnya adalah akurasi sebesar 0.5263157894736842 atau sekitar 52.63%.

```
from sklearn.metrics import accuracy_score
y_true = test_gen.classes
y_pred = model.predict(test_gen)
y_pred = np.argmax(y_pred, axis=1)

# Hitung akurasi
accuracy = accuracy_score(y_true, y_pred)
print("Akurasi: ", accuracy)

1/1 [=====] - 5s 5s/step
Akurasi: 0.5263157894736842
```

Gambar 10. Hasil Model

4. SIMPULAN

Pada penelitian ini, model jaringan saraf tiruan telah dilatih dan dievaluasi menggunakan dataset yang telah dibagi menjadi data pelatihan, data validasi, dan data pengujian dan mendapatkan akurasi model pada data pengujian tercapai sebesar 0.5263 atau sekitar 52.63%.

5. SARAN

Akurasi sebesar 52.63% menunjukkan bahwa model memiliki performa yang cukup baik dalam melakukan prediksi. Namun, tergantung pada kasus penggunaan dan kelas yang dihadapi, angka ini mungkin masih dapat ditingkatkan.

Anda dapat mencoba beberapa teknik untuk meningkatkan performa model, seperti:

- a. Menggunakan arsitektur jaringan yang lebih kompleks atau menggunakan pre-trained model lain yang lebih sesuai dengan dataset yang Anda miliki.
- b. Melakukan penyetelan hyperparameter yang lebih cermat, seperti mengubah learning rate, batch size, atau jumlah epoch yang digunakan dalam pelatihan.
- c. Melakukan augmentasi data untuk memperkaya dataset pelatihan, seperti rotasi, pergeseran, flipping, atau perubahan skala gambar.
- d. Memeriksa kembali dataset untuk memastikan kualitas dan keberagaman yang memadai dalam setiap kelas.

DAFTAR PUSTAKA

- [1] N. D. Girsang, "Classification Of Batik Images Using Multilayer Perceptron With Histogram Of Oriented Gradient Feature Extraction," in *Proceeding International Conference on Science and Engineering*, 2021, pp. 197–204.
- [2] C. Jatmoko and D. Sinaga, "Ekstraksi Fitur Glcm Pada K-Nn Dalam Mengklasifikasi Motif Batik," 2019.
- [3] A. A. Trixie, "Filosofi Motif Batik Sebagai Identitas Bangsa Indonesia," *Folio*, vol. 1, no. 1, pp. 1–9, 2020.
- [4] R. A. Putri and N. Rochmawati, "Penerapan Algoritma Support Vector Machine untuk Klasifikasi Motif Citra Batik Solo Berdasarkan Fitur Multi-Autoencoders," *Journal of Informatics and Computer Science (JINACS)*, vol. 1, no. 01, pp. 56–63, 2019.
- [5] C. Umam and L. B. Handoko, "Convolutional Neural Network (CNN) Untuk Identifikasi Karakter Hiragana," in *Prosiding Seminar Nasional Lppm Ump*, 2021, pp. 527–533.
- [6] S. C. A. Pradhana, U. N. Wisesty, and F. Sthevanie, "Pengenalan Aksara Jawa dengan Menggunakan Algoritma Convolutional Neural Network," *eProceedings of Engineering*, vol. 7, no. 1, 2020.
- [7] M. M. Susilo, D. M. Wonohadidjojo, and N. Sugianto, "Pengenalan Pola Karakter Bahasa Jepang Hiragana Menggunakan 2D Convolutional Neural Network," *Jurnal Informatika dan Sistem Informasi*, vol. 3, no. 2, pp. 28–36, 2017.
- [8] T. A. Bowo, H. Syaputra, and M. Akbar, "Penerapan Algoritma Convolutional Neural Network Untuk Klasifikasi Motif Citra Batik Solo," *Journal of Software Engineering Ampera*, vol. 1, no. 2, pp. 82–96, May 2020, doi: 10.51519/journalsea.v1i2.47.
- [9] M. R. A. Yudianto, K. Kusriani, and H. Al Fatta, "Analisis Pengaruh Tingkat Akurasi Klasifikasi Citra Wayang dengan Algoritma Convolutional Neural Network," (*JurTI*) *Jurnal Teknologi Informasi*, vol. 4, no. 2, pp. 182–191, 2020.
- [10] A. Prayoga, P. Sukmasetya, M. R. A. Yudianto, and R. A. Hasani, "Arsitektur Convolutional Neural Network untuk Model Klasifikasi Citra Batik Yogyakarta," *Journal of Applied Computer Science and Technology*, vol. 4, no. 2, pp. 82–89, 2023.
- [11] K. Wabang and O. D. Nurhayati, "Application of The Naïve Bayes Classifier Algorithm to Classify Community Complaints," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 6, no. 5, pp. 872–876, 2022.
- [12] B. P. Pratiwi, A. S. Handayani, and S. Sarjana, "Pengukuran Kinerja Sistem Kualitas Udara Dengan Teknologi Wsn Menggunakan Confusion Matrix," *Jurnal Informatika Upgris*, vol. 6, no. 2, 2020.