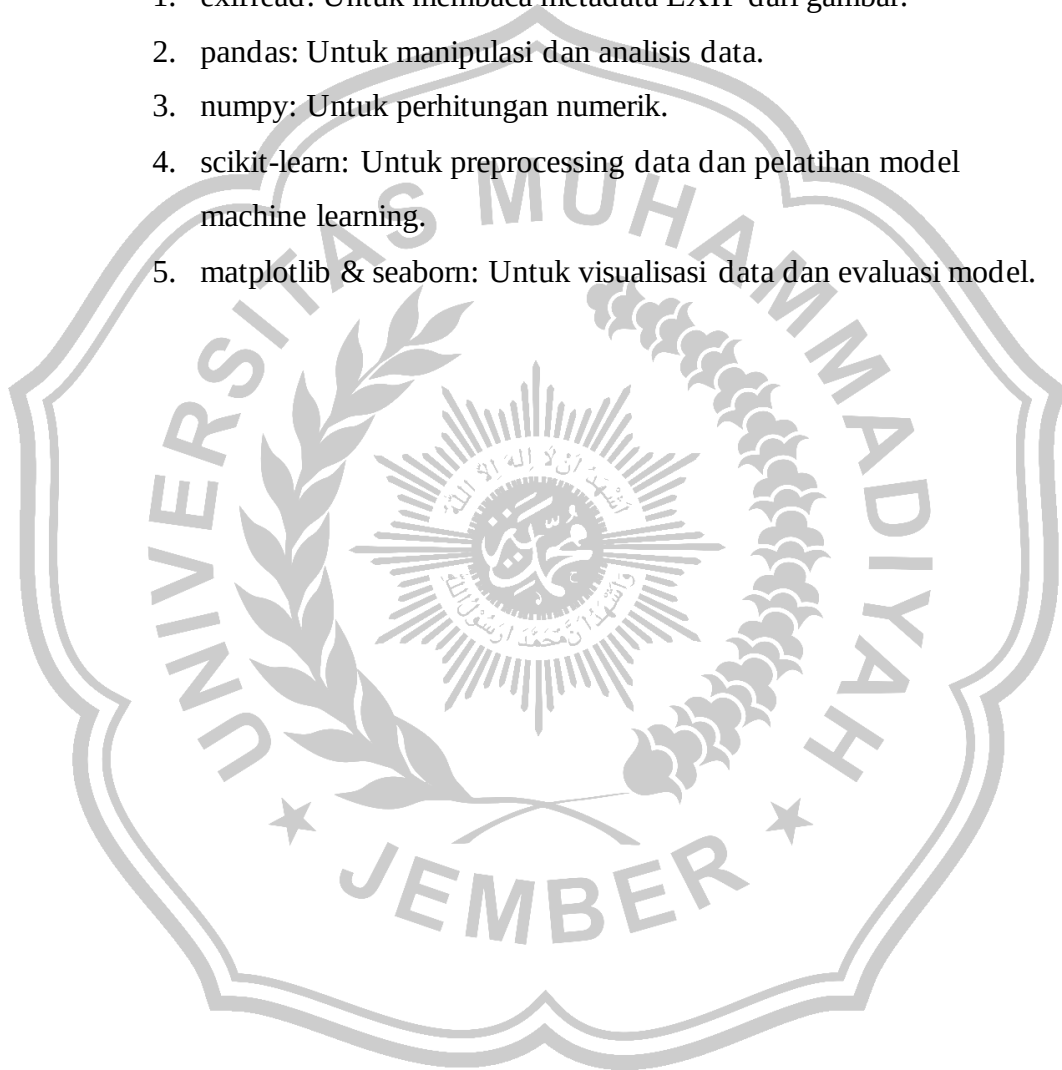


Untuk menjalankan penelitian ini, digunakan beberapa alat dan pustaka sebagai berikut:

1. Google Colab: Platform berbasis cloud untuk menjalankan kode Python dengan dukungan GPU.
2. Python: Bahasa pemrograman utama yang digunakan.
3. Pustaka yang digunakan:
 1. exifread: Untuk membaca metadata EXIF dari gambar.
 2. pandas: Untuk manipulasi dan analisis data.
 3. numpy: Untuk perhitungan numerik.
 4. scikit-learn: Untuk preprocessing data dan pelatihan model machine learning.
 5. matplotlib & seaborn: Untuk visualisasi data dan evaluasi model.



BAB 4

HASIL DAN PEMBAHASAN

4.1. Pembahasan

Bab ini membahas tentang proses pengujian model machine learning yang telah dikembangkan untuk membedakan gambar asli dan gambar AI. Pengujian dilakukan dengan menggunakan dataset yang terdiri dari 80 gambar (40 gambar asli dan 40 gambar AI) serta 10 gambar uji coba. Model yang digunakan telah dilatih dengan teknik supervised learning menggunakan algoritma Random Forest. Pengujian dilakukan untuk mengevaluasi kinerja model dalam mengklasifikasikan gambar berdasarkan metadata seperti merek perangkat (Make), model perangkat (Model), serta informasi GPS (GPSInfo).

4.2. Dataset Pengujian

Dataset yang digunakan dalam penelitian ini terdiri dari 80 gambar yang terbagi menjadi dua kategori, yaitu gambar asli dan gambar yang dibuat menggunakan AI. Selain itu, terdapat 10 gambar tambahan yang digunakan sebagai data uji coba.

Setiap gambar dalam dataset memiliki metadata yang mencakup informasi penting seperti:

1. *Make*: Merek perangkat yang digunakan untuk mengambil gambar (hanya ada pada gambar asli).
2. *Model*: Model perangkat yang digunakan (hanya ada pada gambar asli).
3. *GPS Info*: Informasi koordinat lokasi pengambilan gambar (hanya ada pada gambar asli).

Gambar AI tidak memiliki metadata *Make*, *Model*, dan *GPSInfo*, sehingga model memanfaatkan perbedaan ini untuk melakukan klasifikasi.

4.3. Metode Pengujian

Pengujian dilakukan dengan tahapan sebagai berikut:

1. *Preprocessing Data*: Mengambil metadata dari gambar dan mengubahnya ke dalam format yang bisa digunakan oleh model machine learning.
2. *Ekstraksi Fitur*: Menyimpan hanya tiga fitur utama yaitu *Make*, *Model*,

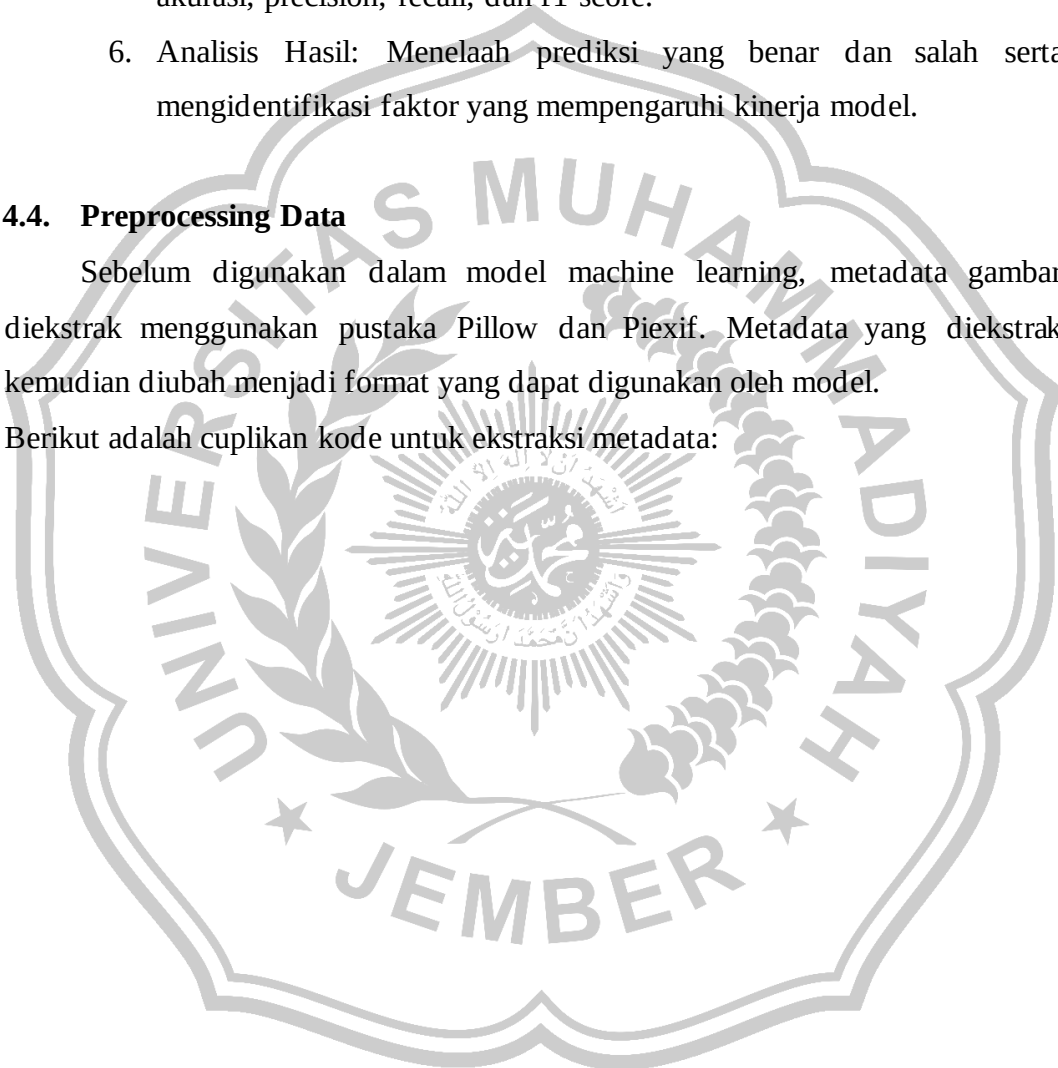
dan GPSInfo.

3. Normalisasi Data: Mengonversi data kategorikal menjadi bentuk numerik agar bisa digunakan dalam model.
4. Pelatihan Model: Melatih model dengan algoritma Random Forest menggunakan dataset pelatihan.
5. Evaluasi Model: Menguji model dengan dataset uji dan menghitung akurasi, precision, recall, dan f1-score.
6. Analisis Hasil: Menelaah prediksi yang benar dan salah serta mengidentifikasi faktor yang mempengaruhi kinerja model.

4.4. Preprocessing Data

Sebelum digunakan dalam model machine learning, metadata gambar diekstrak menggunakan pustaka Pillow dan Piexif. Metadata yang diekstrak kemudian diubah menjadi format yang dapat digunakan oleh model.

Berikut adalah cuplikan kode untuk ekstraksi metadata:



```

from PIL import Image
import piexif
import pandas as pd
import os

# Fungsi untuk ekstrak metadata
def extract_metadata(image_path):
    img = Image.open(image_path)
    exif_data = piexif.load(img.info.get("exif", b""))
    make = exif_data.get('0th', {}).get(piexif.ImageIFD.Make, b'')
    model = exif_data.get('0th', {}).get(piexif.ImageIFD.Model, b'')
    gps = exif_data.get('GPS', {})
    gps_info = "GPS Available" if gps else "No GPS"

    return [make, model, gps_info]

# Membaca dataset
data = []
labels = []
folder_paths = {'asli': 'dataset/asli', 'ai': 'dataset/ai'}

for label, folder in folder_paths.items():
    for filename in os.listdir(folder):
        file_path = os.path.join(folder, filename)
        metadata = extract_metadata(file_path)
        data.append(metadata)
        labels.append(label)

# Menyimpan ke DataFrame
df = pd.DataFrame(data, columns=['Make', 'Model', 'GPS'])
df['Label'] = labels

```

Gambar 4.1 Preprocessing Data

4.5. Pembuatan Model Machine Learning

Setelah metadata diekstrak, dilakukan proses encoding terhadap data kategorikal (Make, Model, GPS) menggunakan **Label Encoding** agar dapat digunakan oleh model **Random Forest**.

Berikut adalah kode untuk pemrosesan data dan pembuatan model:

```
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder, StandardScaler
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, accuracy_score

# Encoding data kategorikal
encoder = LabelEncoder()
df['Make'] = encoder.fit_transform(df['Make'])
df['Model'] = encoder.fit_transform(df['Model'])
df['GPS'] = encoder.fit_transform(df['GPS'])

# Memisahkan fitur dan label
X = df[['Make', 'Model', 'GPS']]
y = df['Label']

# Membagi data menjadi training dan testing
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Membuat model Random Forest
model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X_train, y_train)
```

Gambar 4. 2 Pembuatan Model Machine Learning

4.6. Evaluasi Model

Evaluasi dilakukan menggunakan metrik berikut:

1. **Akurasi:** Mengukur persentase prediksi yang benar dibandingkan dengan total prediksi.
2. **Precision:** Menunjukkan seberapa tepat model dalam mengklasifikasikan gambar asli dan AI.
3. **Recall:** Mengukur sejauh mana model mampu menemukan semua

instance dari kelas tertentu.

4. **F1-Score:** Kombinasi antara precision dan recall yang memberikan

Tabel 4 1 Evaluasi

Metrik	Ai	Asli
Precision	0.86	1.00
Recall	1.00	0.91
F1-Score	0.92	0.95
Akurasi	0.94	

Berdasarkan tabel di atas, model mampu membedakan antara gambar asli dan AI dengan tingkat akurasi **94%**. Hal ini menunjukkan bahwa metode yang digunakan cukup efektif dalam mengklasifikasikan gambar berdasarkan metadata yang tersedia

4.7. Pengujian dengan Data Baru

Untuk menguji model dengan gambar baru, dilakukan proses ekstraksi metadata dan prediksi:

```
# Contoh gambar uji coba
new_image_path = 'test_images/test1.jpg'
new_metadata = extract_metadata(new_image_path)
new_df = pd.DataFrame([new_metadata], columns=['Make', 'Model',

# Encoding dengan model yang sudah dilatih
new_df['Make'] = encoder.transform(new_df['Make'])
new_df['Model'] = encoder.transform(new_df['Model'])
new_df['GPS'] = encoder.transform(new_df['GPS'])

# Prediksi dengan model
prediction = model.predict(new_df)
print(f' 🚀 Prediksi: {prediction[0]}')
```

Gambar 4. 3 Pengujian dengan Data Baru

4.8. Hasil Pengujian

Model diuji dengan 10 gambar uji yang terdiri dari 5 gambar asli dan 5 gambar AI. Berikut adalah hasil klasifikasinya:

Tabel 4 2 Hasil Pengujian

No	Nama File	Kelas aktual	Prediksi	Hasil
1	testing asli (1).jpg	Asli	Asli	Benar
2	testing asli (2).jpg	Asli	Ai	Salah
3	testing asli (3).jpg	Asli	Asli	Benar
4	testing asli (4).jpg	Asli	Asli	Benar
5	testing asli (5).jpg	Asli	Asli	Benar
6	testing (1) AI .jpeg	Ai	Ai	Benar
7	testing (2) AI .jpeg	Ai	Ai	Benar
8	testing (3) AI .jpeg	Ai	Ai	Benar
9	testing (4) AI .jpeg	Ai	Ai	Benar
10	testing (5) AI .jpeg	Ai	Ai	Benar

4.9. Analisi Kesalahan

Dari hasil pengujian, terdapat satu gambar yang salah diklasifikasikan sebagai "Asli" padahal gambar tersebut merupakan hasil AI. Kemungkinan penyebab kesalahan ini antara lain:

1. Metadata gambar hasil AI mungkin telah dimodifikasi agar menyerupai foto asli.
2. Model masih memiliki keterbatasan dalam mengenali pola metadata tertentu.
3. Beberapa metadata yang digunakan mungkin memiliki nilai yang tumpang tindih antara gambar asli dan gambar AI.