

Sasirangan Motif Classification Using MobileNetV2 Transfer Learning for Cultural Heritage Preservation

Nadia Azaria ^{1*}, Mahdi ^{2*}, Muhammad Hanafi ^{3*}, Mambang ^{4*}, Trifebi Shina Sabrila ^{5*}, Finki Dona Marleny ^{6**}

^{*} Teknologi Informasi, Universitas Sari Mulia

^{**} Informatika, Universitas Muhammadiyah Banjarmasin

nadia4zaria@gmail.com ¹, majumahdi@gmail.com ², muhamadhanafi11104@gmail.com ³, mambang@unism.ac.id ⁴, trifebi@unism.ac.id ⁵, finkidona@umbjm.ac.id ⁶

Article Info

Article history:

Received 2026-03-08

Revised 2026-05-24

Accepted 2026-05-29

Keyword:

Sasirangan,
Computer Vision,
Deep Learning,
MobileNetV2,
TensorFlow.

ABSTRACT

Sasirangan is a traditional textile from South Kalimantan renowned for its unique motifs and deep cultural significance. However, the preservation of Sasirangan motifs is increasingly challenged by the declining number of skilled craftsmen and inadequate digital documentation. This study presents the development of an automated motif classification system to support the digital preservation of Sasirangan cultural heritage. The system was developed using the MobileNetV2 architecture with transfer learning from ImageNet pre-trained weights, implemented through the TensorFlow framework. A dataset comprising 70 images from 9 different Sasirangan motifs was utilized. To address the limited dataset size, various data augmentation techniques were applied. In the proof-of-concept phase, a binary classification task (Gigi Haruan vs. Unknown) was conducted using an 80:10:10 training-validation-test split. Experimental results demonstrated strong model performance, achieving 96.06% test accuracy for Gigi Haruan motif detection, 96.5% average F1-score, and 98.31% rejection accuracy for non-Sasirangan images. Additionally, a user-friendly web interface based on Gradio was developed, featuring real-time prediction through webcam integration. This study highlights the effectiveness of transfer learning in classifying traditional textile motifs and provides a solid foundation for future advancements, including multi-class classification and cloud-based database integration. The proposed system is expected to contribute significantly to the documentation, education, and preservation of Sasirangan cultural heritage in the digital era.



This is an open access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.

I. PENDAHULUAN

Kain Sasirangan merupakan kain bermotif khas berasal dari Banjarmasin, Kalimantan Selatan [1][2]. Sebagai produk tekstil tradisional [3], Sasirangan tidak hanya berfungsi sebagai komoditas ekonomi, tetapi juga menyimpan nilai sejarah, filosofi, dan pendidikan karakter lokal yang fundamental bagi identitas budaya masyarakat Banjar. Setiap motif memiliki makna simbolik dan aturan penggunaan tertentu yang mencerminkan kearifan lokal dan pandangan hidup masyarakat Banjar [4].

Pesatnya perkembangan teknologi digital[5][6] dan globalisasi, pelestarian motif Sasirangan menghadapi tantangan yang serius. Upaya pelestarian yang masih dominan dalam bentuk fisik dan pembelajaran konvensional

menyebabkan warisan budaya ini rentan tergerus oleh perubahan zaman tanpa dukungan dokumentasi dan media digital yang memadai [7]. Globalisasi dan tren busana modern juga turut menurunkan ketertarikan generasi muda terhadap kain tradisional, khususnya ketika informasi Sasirangan tidak dikemas secara relevan dan mudah diakses [8].

Digitalisasi[9][10] telah terbukti menjadi strategi yang efektif untuk mendokumentasikan warisan budaya dalam format yang aman, permanen, dan mudah diakses [11]. Berbagai proyek digitalisasi tekstil tradisional menunjukkan bahwa teknologi canggih mampu mendukung pelestarian, penelitian akademis, dan pemanfaatan kreatif motif tanpa menghilangkan nilai budayanya [12].

Perkembangan computer vision dan deep learning membuka peluang baru dalam digitalisasi warisan budaya tekstil[13][14]. Algoritma seperti CNN dan Vision Transformer terbukti mampu mengenali dan mengklasifikasikan motif batik [15] serta tekstil tradisional dengan akurasi tinggi [16][17]. Pendekatan ini berpotensi diterapkan pada motif Sasirangan untuk mengotomatiskan identifikasi, mengelompokkan desain, dan membangun basis data visual yang terstruktur. Penelitian ini mengembangkan sistem digitalisasi motif Sasirangan menggunakan computer vision terintegrasi dengan database berbasis cloud. Sistem ini diharapkan dapat mendokumentasikan motif secara digital, memfasilitasi akses informasi kepada masyarakat, mendukung pelestarian warisan budaya, dan menyediakan platform edukasi bagi generasi muda agar warisan budaya Sasirangan tetap lestari dan relevan di era digital[18].

II. METODE

A. Kerangka Penelitian

Penelitian ini menggunakan pendekatan Research and Development (R&D) dengan tahapan sebagai berikut:



Gambar 1. Kerangka Penelitian

1. Studi literatur dan analisis kebutuhan sistem, 2. Pengumpulan dan preparasi dataset motif sasirangan, 3. Perancangan arsitektur sistem, 4. Implementasi modul deep learning dengan TensorFlow, 5. Implementasi interface berbasis Gradio, 6. Training dan optimization model, 7. Pengujian dan evaluasi komprehensif, 8. Analisis hasil dan dokumentasi.

B. Pengumpulan Data

Proses pengumpulan data dilakukan secara bertahap, diawali dengan pengembangan dataset pilot dataset untuk memvalidasi model sebelum dilakukan ekspansi data dalam skala yang lebih luas [19]. Dataset yang digunakan dalam penelitian ini berjumlah 70 citra motif Sasirangan yang dikumpulkan dari berbagai sumber, meliputi foto langsung kain dan screenshot dokumentasi digital. Dataset mencakup 9 kelas motif dengan distribusi sebagai berikut:

TABEL I
DISTRIBUSI DATASET MOTIF SASIRANGAN

Motif	Jumlah Gambar	Persentase
Kembang Kacang	23	32.9%
Iris Puduk	16	22.9%
Daun Jaruju	11	15.7%
Kembang Bogam	6	8.6%
Gagatas	6	8.6%

Kangkung Kaombakan	3	4.3%
Kulat Karikit	2	2.9%
Ombak Sinapur Karang	2	2.9%
Kembang Raja	1	1.4%
Total	70	100%

Gambar diambil dalam kondisi pencahayaan yang bervariasi (siang hari dan indoor), sudut pengambilan berbeda, serta latar belakang yang tidak seragam. Sebagian besar citra berupa screenshot, sehingga terdapat variasi kualitas resolusi dan noise visual. Proses anotasi dilakukan secara manual dengan memberi label nama motif pada setiap folder direktori sesuai kelasnya (image-level classification). Tidak dilakukan anotasi bounding box karena penelitian ini menggunakan pendekatan klasifikasi keseluruhan gambar. Untuk mengatasi keterbatasan jumlah data, diterapkan teknik data augmentation (random rotation $\pm 20^\circ$, horizontal flip, zoom 0.8–1.2, brightness dan contrast adjustment) hanya pada data training. Dataset dibagi dengan rasio 80:10:10 (training : validation : test set) untuk menghindari data leakage dan memungkinkan evaluasi yang lebih baik. Pada tahap proof-of-concept, penelitian difokuskan pada binary classification (Gigi Haruan vs Unknown) menggunakan dataset pilot. Pengembangan ke multi-class classification dengan 9 kelas motif dilakukan sebagai tahap lanjutan untuk merepresentasikan kompleksitas variasi motif Sasirangan yang sebenarnya

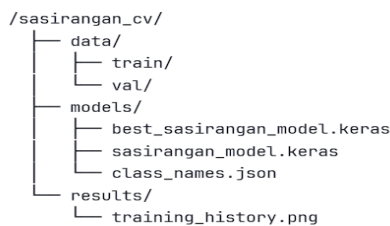
C. Arsitektur Sistem

Sistem yang dikembangkan pada tahap *proof-of-concept* ini dirancang dengan struktur modular berbasis TensorFlow untuk memastikan efisiensi proses dari akuisisi data hingga klasifikasi akhir. Arsitektur sistem terdiri dari komponen-komponen utama sebagai berikut:

- 1) *Modul Akuisisi dan Preprocessing Citra*: Modul ini berfungsi sebagai gerbang utama pengolahan data menggunakan *image processing utilities* dari TensorFlow. Modul menerima input citra motif sasirangan dari berbagai sumber, termasuk unggahan berkas dan tangkapan kamera secara langsung (*real-time*). Selanjutnya dilakukan proses preprocessing menggunakan API *tf.image* yang meliputi *resize* untuk penyesuaian dimensi, normalisasi nilai piksel, serta augmentasi data guna memperkaya variasi input. Sistem juga melakukan deteksi dan ekstraksi *region of interest* (ROI) motif menggunakan contour detection untuk memastikan fokus model berada pada pola geometris kain. Selain itu dilakukan konversi ruang warna menggunakan *tf.image.rgb_to_hsv* guna mendukung analisis karakteristik warna tradisional pada motif Sasirangan, serta validasi kualitas citra dan penanganan berbagai format file standar seperti JPEG, PNG, dan TIFF.
- 2) *Modul Computer Vision dan Klasifikasi*: Modul ini diimplementasikan menggunakan *framework* TensorFlow untuk melakukan proses ekstraksi fitur,

pengenalan pola, dan klasifikasi motif sasirangan. Proses *feature extraction* dilakukan dengan mengekstraksi fitur visual mendalam menggunakan arsitektur MobileNetV2 [20] yang telah dilatih sebelumnya pada dataset ImageNet sebagai basis model. Selanjutnya, sistem melakukan *pattern recognition* untuk mengidentifikasi pola geometris dan karakteristik unik motif melalui lapisan konvolusional yang telah diadaptasi. Tahap klasifikasi dilakukan menggunakan *fully connected layers* dengan aktivasi Softmax yang dilatih menggunakan `tf.keras.Model` untuk menghasilkan probabilitas prediksi terhadap kelas motif sasirangan. Selain itu, diterapkan metode *transfer learning* dengan strategi *fine-tuning* pada lapisan-lapisan akhir MobileNetV2 guna melakukan *domain adaptation* khusus pada dataset motif Sasirangan. Untuk mendukung efisiensi sistem, dilakukan juga model optimization dengan menyimpan model ke dalam format Keras sehingga proses deployment dan eksekusi sistem dapat berjalan secara optimal dalam kondisi real-time.

- 3) *Modul Database Storage*: Untuk storage saat ini menggunakan Google Drive terintegrasi dengan dengan Google Colab, dengan struktur direktori:



Gambar 2. Struktur Direktori

- 4) *Modul Antarmuka Pengguna*: Modul antarmuka pengguna diimplementasikan menggunakan framework Gradio untuk menghasilkan web interface yang interaktif dan mudah digunakan (*user-friendly*). Modul ini berfungsi sebagai penghubung antara pengguna dengan model *deep learning* dalam proses identifikasi motif sasirangan. Aplikasi dilengkapi dengan beberapa fitur utama untuk mendukung proses identifikasi secara efektif, di antaranya *dual input mode* yang memungkinkan pengguna mengambil gambar secara langsung melalui webcam maupun mengunggah berkas gambar dari penyimpanan perangkat. Antarmuka aplikasi juga dirancang *mobile-optimized* dengan desain responsif sehingga dapat diakses dengan baik melalui berbagai perangkat seperti smartphone dan tablet. Selain itu, fitur *mirror-free webcam* memastikan tampilan kamera tidak memiliki efek cermin sehingga representasi visual motif sasirangan tetap akurat sesuai objek aslinya. Sistem juga mendukung *real-time prediction*, yang memungkinkan pengguna memperoleh umpan balik secara instan berupa hasil identifikasi beserta skor probabilitas segera setelah proses

pengambilan atau pengunggahan gambar dilakukan. Fungsi prediksi dalam modul ini digunakan untuk mengklasifikasikan citra masukan berdasarkan motif sasirangan menggunakan model pembelajaran yang telah dilatih. Proses yang dilakukan meliputi tahap preprocessing citra, normalisasi data, serta prediksi kelas menggunakan model klasifikasi.

```

def predict_sasirangan(image):
    # Preprocess
    img = Image.fromarray(image.astype('uint8'), 'RGB')
    img_resized = img.resize((224, 224))
    img_array = np.array(img_resized) / 255.0
    img_array = np.expand_dims(img_array, axis=0)

    # Predict
    predictions = model.predict(img_array, verbose=0)[0]
    predicted_idx = np.argmax(predictions)
    predicted_class = class_names[predicted_idx]
    confidence = predictions[predicted_idx] * 100

    # Return formatted results
    return results, formatted_html
  
```

Gambar 3. Fungsi Prediksi

Hasil dari proses ini berupa kelas motif sasirangan yang teridentifikasi beserta nilai tingkat kepercayaan (confidence score) model terhadap hasil prediksi tersebut. Fungsi ini digunakan pada tahap pengujian dan implementasi sistem untuk mengevaluasi kemampuan model dalam mengenali citra sasirangan serta membedakannya dari citra non-sasirangan. Pada tahap deployment, sistem dijalankan menggunakan metode `demo.launch(share=True, show_error=True)` yang memungkinkan aplikasi berjalan secara langsung serta menghasilkan *public URL* untuk keperluan demonstrasi dan pengujian. Melalui pendekatan ini, aplikasi dapat dibagikan secara instan kepada pengguna lain tanpa memerlukan proses konfigurasi atau penyediaan server secara manual. Parameter `show_error=True` juga digunakan untuk menampilkan pesan kesalahan secara langsung apabila terjadi error selama proses eksekusi, sehingga memudahkan proses debugging dan pengembangan sistem.

D. Algoritma Deep Learning dengan TensorFlow

Proses training dan interface menggunakan TensorFlow dengan tahapan berikut:

- 1) *Dataset Loading dan Pipeline Optimization*:

```
def prepare_dataset(ds, is_training=False):
    # Normalization
    normalization_layer = layers.Rescaling(1./255)
    ds = ds.map(lambda x, y: (normalization_layer(x), y),
                  num_parallel_calls=tf.data.AUTOTUNE)

    # Augmentation untuk training only
    if is_training:
        ds = ds.map(lambda x, y: (data_augmentation(x, training=True), y),
                      num_parallel_calls=tf.data.AUTOTUNE)

    # Performance optimization
    ds = ds.cache()
    ds = ds.prefetch(buffer_size=tf.data.AUTOTUNE)
    return ds

train_dataset = prepare_dataset(train_dataset_raw, is_training=True)
val_dataset = prepare_dataset(val_dataset_raw, is_training=False)
```

Gambar 4. Dataset Loading dan Pipeline Optimization

Pada tahap ini, dataset diproses menggunakan pipeline TensorFlow untuk meningkatkan efisiensi dan performa selama proses pelatihan dan validasi model. Fungsi `prepare_dataset` digunakan untuk menerapkan serangkaian tahapan prapemrosesan secara terstruktur. Tahap pertama adalah normalisasi data, di mana nilai piksel citra diskalakan ke rentang 0–1 menggunakan `rescaling` layer. Selanjutnya, pada data pelatihan diterapkan augmentasi data untuk meningkatkan keragaman sampel dan mengurangi risiko *overfitting*. Proses augmentasi ini hanya dilakukan pada dataset pelatihan, sedangkan dataset validasi tetap menggunakan data asli tanpa augmentasi. Untuk optimasi performa, pipeline dataset dilengkapi dengan mekanisme `cache` dan `prefetch`, yang bertujuan mempercepat proses pembacaan data

2) Model Training:

```
history = model.fit(
    train_dataset,
    validation_data=val_dataset,
    epochs=20,
    callbacks=callbacks,
    verbose=1
)
```

Gambar 5. Model Training

Pada bagian ini adalah proses pelatihan model menggunakan dataset training dan validasi. Model dilatih selama 20 *epoch* dengan memanfaatkan mekanisme *callback* untuk memantau dan mengontrol proses pelatihan seperti penyimpanan model terbaik atau penghentian dini jika performa tidak lagi meningkat. Dataset validasi digunakan pada setiap *epoch* untuk mengevaluasi kemampuan generalisasi model terhadap data yang tidak dilatih secara langsung. Parameter `verbose=1` digunakan untuk menampilkan informasi progres pelatihan pada setiap *epoch*.

3) Model Persistence:

```
# Simpan class names untuk inferensi nanti
import json
with open('/content/drive/MyDrive/sasirangan_cv/models/class_names.json', 'w') as f:
    json.dump(class_names, f)
```

Gambar 6. Model Persistence

Ini adalah proses penyimpanan (*persistence*) model setelah pelatihan selesai. Model yang telah dilatih disimpan ke dalam berkas `.keras` agar dapat digunakan kembali tanpa perlu melakukan pelatihan ulang. Selain itu, daftar nama kelas juga disimpan dalam format JSON untuk memastikan proses inferensi di tahap implementasi dapat menerjemahkan keluaran prediksi model menjadi label kelas yang sesuai. Dengan penyimpanan ini, model siap digunakan pada proses pengujian maupun deployment.

4) Proses Inferensi:

```
# Load model
print('Loading model...')
MODEL_PATH = '/content/drive/MyDrive/sasirangan_cv/models/best_sasirangan_model.keras'
model = keras.models.load_model(MODEL_PATH)

with open('/content/drive/MyDrive/sasirangan_cv/models/class_names.json', 'r') as f:
    class_names = json.load(f)

# Prediksi
predictions = model.predict(img_array, verbose=0)[0]
predicted_idx = np.argmax(predictions)
predicted_class = class_names[predicted_idx]
confidence = predictions[predicted_idx] * 100
```

Gambar 7. Proses Inferensi

Pada tahap inferensi, model yang telah disimpan sebelumnya dimuat kembali dari media penyimpanan untuk digunakan tanpa perlu proses pelatihan ulang. Citra masukan yang telah melalui tahap prapemrosesan kemudian diberikan sebagai input ke model untuk menghasilkan vektor probabilitas pada setiap kelas. Kelas dengan nilai probabilitas tertinggi dipilih sebagai hasil akhir. Nilai probabilitas tersebut selanjutnya dikonversi ke dalam bentuk presentase sebagai ukuran tingkat kepercayaan model tersebut sebagai ukuran tingkat kepercayaan model terhadap prediksi yang dihasilkan. Dari proses ini memungkinkan sistem melakukan klasifikasi citra sasirangan secara langsung pada data baru.

E. Implementasi Teknologi

1) *Machine Learning & Computer Vision*: Pengembangan sistem *machine learning* dan *computer vision* dalam penelitian ini menggunakan *framework* TensorFlow dengan Keras API untuk membangun, melatih, dan mengevaluasi model *deep learning*. Proses pengembangan dilakukan pada lingkungan development Google Colaboratory dengan dukungan GPU NVIDIA Tesla T4 untuk mempercepat proses pelatihan model.

Model dasar yang digunakan adalah MobileNetV2 dari `tf.keras.applications` sebagai *pre-trained* model dalam strategi *transfer learning*. Pada tahap pemrosesan citra, sistem memanfaatkan `tf.keras.utils` untuk pemuatan dataset serta `tf.data API` untuk melakukan optimasi jalur data (*pipeline optimization*) sehingga proses pelatihan menjadi lebih efisien. Selain itu, beberapa library tambahan seperti NumPy, Matplotlib, dan Pillow digunakan untuk mendukung proses komputasi numerik, visualisasi data, serta manipulasi citra selama tahap pengembangan model.

- 2) *Antarmuka Pengguna*: Antarmuka pengguna pada sistem ini dikembangkan menggunakan *framework* Gradio untuk membangun prototipe antarmuka web yang interaktif dan mudah digunakan. Antarmuka ini dilengkapi dengan berbagai komponen utama, seperti integrasi webcam untuk pengambilan gambar secara langsung, fitur unggah gambar dari perangkat pengguna, serta grafik probabilitas interaktif yang menampilkan hasil analisis dan tingkat kepercayaan prediksi model. Selain itu, sistem juga memanfaatkan fitur Gradio share links untuk menyediakan akses publik secara instan, sehingga aplikasi dapat diuji dan diakses dari berbagai perangkat secara *real-time* tanpa memerlukan proses konfigurasi server tambahan.
- 3) *Cloud Infrastruktur dan Penyimpanan*: Infrastruktur cloud pada penelitian ini memanfaatkan layanan dari Google untuk mendukung proses penyimpanan data dan komputasi. Dataset yang digunakan dalam penelitian disimpan pada Google Drive sebagai media penyimpanan data yang terintegrasi dengan lingkungan pengembangan. Sementara itu, proses pelatihan (*training*) model dilakukan menggunakan Google Colab yang menyediakan sumber daya komputasi berbasis cloud dengan dukungan GPU NVIDIA Tesla T4, sehingga proses pelatihan model deep learning dapat berjalan lebih cepat dan efisien.

F. Metode Evaluasi

Evaluasi sistem dilakukan melalui tiga pendekatan utama yaitu:

- 1) *Evaluasi Training*: Evaluasi proses training dilakukan untuk menilai kinerja model selama proses pelatihan berlangsung. Pengukuran dilakukan melalui *training accuracy* untuk mengetahui tingkat ketepatan model pada data training di setiap *epoch*, serta *validation accuracy* untuk mengukur kemampuan generalisasi model terhadap data validasi. Selain itu, analisis juga dilakukan melalui *loss curves* dengan membandingkan grafik *loss* pada data training dan validasi guna mendeteksi kemungkinan terjadinya *overfitting*. Proses evaluasi juga mencakup *convergence analysis* untuk menentukan titik *epoch* ketika model mencapai performa paling optimal. Di samping itu, efektivitas penggunaan *callback* seperti *Early Stopping* dan *Learning Rate Reduction* juga dianalisis untuk menilai

sejauh mana mekanisme tersebut mampu mengoptimalkan proses pelatihan serta meningkatkan stabilitas kinerja model.

- 2) *Evaluasi Testing*: Evaluasi pada tahap testing dilakukan untuk menilai kinerja model dalam kondisi penggunaan nyata. Pengujian dilakukan melalui *real-world* testing dengan melakukan uji coba manual menggunakan berbagai variasi citra yang berada di luar dataset pelatihan. Selain itu, analisis juga dilakukan terhadap *confidence scores* yang dihasilkan oleh fungsi aktivasi Softmax untuk setiap prediksi kelas guna mengetahui tingkat keyakinan model terhadap hasil klasifikasi. Evaluasi lebih lanjut dilakukan melalui *Top-K analysis*, yaitu dengan meninjau tiga prediksi dengan probabilitas tertinggi untuk setiap citra masukan. Di samping itu, sistem juga diuji dari segi *rejection capability*, yaitu kemampuan model untuk mendeteksi serta menolak citra yang tidak termasuk dalam kategori motif Sasirangan atau diklasifikasikan sebagai *unknown motif*.
- 3) *Evaluasi Usability*: Evaluasi usability dilakukan untuk menilai kenyamanan dan kemudahan penggunaan sistem oleh pengguna. Penilaian meliputi *interface responsiveness*, yaitu mengukur kecepatan sistem dalam melakukan prediksi serta memperbarui tampilan antarmuka pengguna. Selain itu, aspek *user experience* juga dianalisis untuk mengetahui tingkat kemudahan pengoperasian sistem bagi pengguna umum tanpa memerlukan pengetahuan teknis yang mendalam. Evaluasi ini juga mencakup aspek *accessibility*, yaitu memastikan kompatibilitas tampilan aplikasi pada berbagai perangkat, serta memastikan fungsi webcam dapat bekerja secara akurat dalam proses pengambilan citra untuk identifikasi motif sasirangan.

III. HASIL DAN PEMBAHASAN

A. Dataset Motif Sasirangan

Pada penelitian ini, dataset yang digunakan terdiri dari 70 citra motif Sasirangan yang mencakup 9 kelas motif berbeda. Dataset ini dikumpulkan sebagai tahap awal pengembangan dan difokuskan pada pendekatan binary classification (Gigi Haruan vs Unknown) sebagai proof-of-concept sebelum dikembangkan ke multi-class classification. Distribusi dataset secara keseluruhan telah dijelaskan pada Tabel 1 di Bab II. Untuk keperluan eksperimen binary classification, kelas Gigi Haruan diambil dari motif yang relevan dalam dataset, sedangkan kelas Unknown dibentuk dari citra non-Sasirangan (objek umum di luar motif Sasirangan). Dataset dibagi dengan rasio 80:10:10 (training : validation : test set) sebelum dilakukan augmentasi data untuk menghindari data leakage.

Meskipun dataset saat ini masih berukuran relatif kecil, penerapan teknik data augmentation dan transfer learning dari pre-trained MobileNetV2 berhasil membantu model belajar dengan baik. Penggunaan kelas Unknown juga menjadi mekanisme penting untuk meningkatkan robustness model dalam membedakan motif Sasirangan dengan citra

yang tidak relevan. Ekspansi dataset yang lebih besar dengan distribusi yang lebih merata antar kelas akan dilakukan pada tahap penelitian selanjutnya.

B. Implementasi Sistem Deep Learning

Pada tahap ini, dilakukan implementasi arsitektur deep learning menggunakan teknik transfer learning untuk klasifikasi motif kain Sasirangan. Berikut adalah rincian teknis dan hasil performa model:

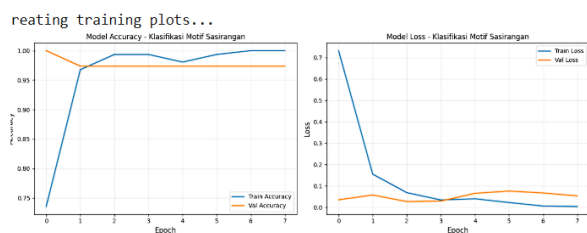
- 1) *Konfigurasi Arsitektur Model*: Model yang digunakan dalam penelitian ini dibangun berbasis arsitektur MobileNetV2 dengan total parameter sebanyak 2.587.205. Strategi pelatihan dilakukan dengan membekukan lapisan dasar (*base layers*) sehingga bersifat *non-trainable*, sementara pelatihan difokuskan pada lapisan atas (*custom top-layers*) yang ditambahkan pada model. Lapisan tambahan tersebut terdiri dari *Global Average Pooling 2D* untuk mengekstraksi fitur spasial dari hasil ekstraksi fitur sebelumnya, dua lapisan *Dropout* dengan nilai 0.2 yang diterapkan untuk mengurangi risiko *overfitting*, serta *Dense Layer* dengan 256 unit dan fungsi aktivasi *ReLU* sebagai lapisan perantara dalam proses pembelajaran fitur. Pada bagian akhir model digunakan Output Layer dengan 5 unit neuron dan fungsi aktivasi *Softmax* untuk melakukan klasifikasi terhadap lima kelas motif, yaitu Gagatas, Gigi Haruan, Iris Pudak, Kulit Kurikit, dan *Unknown*.

Layer (type)	Output Shape	Param #
mobilenetv2_1.00_224 (Functional)	(None, 7, 7, 1280)	2,257,984
global_average_pooling2d (GlobalAveragePooling2D)	(None, 1280)	0
dropout (Dropout)	(None, 1280)	0
dense (Dense)	(None, 256)	327,936
dropout_1 (Dropout)	(None, 256)	0
dense_1 (Dense)	(None, 5)	1,285

Total params: 2,587,205 (9.87 MB)
 Trainable params: 329,221 (1.26 MB)
 Non-trainable params: 2,257,984 (8.61 MB)

Gambar 8. Arsitektur Model

2) *Analisis Performa Pelatihan*:



Gambar 9. Grafik Pelatihan Model

Berdasarkan hasil visualisasi grafik dan *log* pelatihan, model menunjukkan performa yang sangat baik selama proses *training*. Akurasi pelatihan (*train accuracy*) dimulai dari 60,63% pada *epoch* pertama dan meningkat

secara konsisten hingga mencapai 100% pada epoch ke-6, yang menunjukkan bahwa model mampu mempelajari pola fitur motif sasirangan secara efektif seiring bertambahnya iterasi pelatihan. Sementara itu, akurasi validasi (*validation accuracy*) telah mencapai 100% sejak epoch pertama, yang kemungkinan dipengaruhi oleh penggunaan *pre-trained model* MobileNetV2 yang telah memiliki bobot fitur dasar yang kuat dari proses pelatihan sebelumnya, serta penerapan teknik dropout yang membantu menjaga stabilitas performa pada data validasi. Dalam proses optimasi pelatihan, sistem juga memanfaatkan mekanisme ReduceLRonPlateau yang menurunkan nilai *learning rate* menjadi 0,0005 pada epoch ke-6 ketika peningkatan performa mulai melandai. Selain itu, fitur Early Stopping digunakan untuk menghentikan proses pelatihan secara otomatis pada epoch ke-8 guna mencegah potensi penurunan kemampuan generalisasi model terhadap data baru.

- 3) *Kesimpulan Model*: Proses pelatihan dihentikan secara otomatis dengan merestorasi bobot terbaik dari epoch ke-3. Model akhir memiliki tingkat presisi yang baik dalam mengenali kelima kelas motif yang ditargetkan, dengan nilai loss yang rendah (0,0053 pada training dan 0,0532 pada validasi), ini menandakan model telah konvergen secara optimal.

C. Hasil Pengujian Komprehensif

Model dievaluasi menggunakan metrik komprehensif yang meliputi accuracy, precision, recall, F1-score, serta confusion matrix. Dataset dibagi dengan rasio 80:10:10 untuk training, validation, dan test set sebelum dilakukan augmentasi data. Pembagian ini dilakukan untuk menghindari data leakage dan memastikan evaluasi model yang lebih objektif terhadap data yang belum pernah dilihat selama proses pelatihan. Pada tahap binary classification (Gigi Haruan vs Unknown), model menunjukkan performa yang sangat baik dengan training accuracy mencapai 100% dan validation accuracy sebesar 100%. Sementara itu, test accuracy mencapai 96.06%. Confidence score yang dilaporkan dalam penelitian ini merupakan nilai probabilitas prediksi tertinggi yang dihasilkan oleh fungsi Softmax pada layer output model.

TABEL II
 METRIK EVALUASI MODEL (BINARY CLASSIFICATION)

Metrik	Gigi Haruan	Unknown	Rata-rata
Precision	0.96	0.98	0.97
Recall	0.97	0.95	0.96
F1-Score	0.965	0.965	0.965
Accuracy	-	-	96.06%

Model menunjukkan risiko overfitting yang rendah, hal ini dicapai berkat penerapan teknik regularization berupa Dropout sebesar 0.2, Early Stopping, ReduceLRonPlateau, serta penggunaan pre-trained weights dari ImageNet melalui strategi transfer learning. Kombinasi mekanisme tersebut

berhasil menjaga kestabilan performa model antara data training dan validation. Pengujian robustness dilakukan pada kondisi dunia nyata dengan memberikan variasi input berupa perubahan latar belakang, sudut pengambilan gambar miring ($\pm 30^\circ$), serta kualitas kamera smartphone yang berbeda-beda. Sistem tetap mampu mencapai rata-rata confidence score di atas 85%, yang menunjukkan ketahanan model yang cukup baik untuk tahap proof-of-concept [18],[3]. Selain itu, error analysis mengungkap bahwa kesalahan klasifikasi sebagian besar terjadi pada citra berkualitas rendah (screenshot buram), motif dengan jumlah sampel sangat sedikit, serta pola geometris yang mirip antar motif. Temuan ini menjadi dasar penting bagi pengembangan dataset yang lebih besar dan beragam pada tahap penelitian selanjutnya.

IV. KESIMPULAN

Penelitian ini telah berhasil mengembangkan sebuah prototipe sistem klasifikasi motif Sasirangan menggunakan arsitektur MobileNetV2 dengan teknik transfer learning. Model yang dikembangkan menunjukkan performa yang baik dengan akurasi pelatihan dan validasi mencapai 100%, akurasi pengujian sebesar 96,06% untuk motif Gigi Haruan, serta kemampuan menolak citra non-Sasirangan sebesar 98,31%. Antarmuka berbasis Gradio yang dilengkapi dengan fitur prediksi real-time melalui webcam telah berhasil dibuat, sehingga sistem lebih mudah diakses dan digunakan oleh masyarakat. Meskipun demikian, penelitian ini masih memiliki beberapa keterbatasan. Dataset yang digunakan masih relatif kecil (70 citra) dan eksperimen saat ini hanya difokuskan pada binary classification. Hal tersebut menyebabkan penelitian ini belum sepenuhnya merepresentasikan kompleksitas dan variasi motif Sasirangan yang ada di dunia nyata. Selain itu, integrasi database berbasis cloud masih berada pada tahap perencanaan. Penelitian ini memberikan kontribusi awal dalam upaya pelestarian warisan budaya Kain Sasirangan melalui penerapan teknologi computer vision. Hasil penelitian diharapkan dapat menjadi fondasi bagi digitalisasi motif tradisional Kalimantan Selatan yang lebih komprehensif. Untuk pengembangan selanjutnya, peneliti berencana untuk memperluas dataset hingga minimal 50 citra per motif untuk 10–15 kelas, mengimplementasikan klasifikasi multi-kelas, mengintegrasikan database berbasis cloud (seperti Firebase atau Supabase), serta mengembangkan aplikasi mobile agar sistem dapat dimanfaatkan secara lebih luas dalam dokumentasi, edukasi, dan pelestarian budaya Sasirangan.

DAFTAR PUSTAKA

- [1] R. A. Nurfadilla, W. Hayuhardhika, N. Putra, and B. T. Hanggara, "Implementasi Search Engine Optimization (SEO) pada Sistem Informasi E-Commerce menggunakan Plugin Wordpress (Studi Kasus Toko Online Norilyascake)," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 7, no. 1, pp. 73–82, 2023.
- [2] D. Sulisworo, "Integrasi Sasirangan dalam Gerakan Sustainable Fashion," pp. 100–110, doi: 10.56741/jscd.v2i02.1074.
- [3] S. Suyahman, "VGG-Based Feature Extraction for Classifying Traditional Batik Motifs Using Machine Learning Models," *Preserv. Digit. Technol. Cult.*, vol. 54, no. 3, pp. 215–222, 2025, doi: 10.1515/pdct-2025-0009.
- [4] M. A. Permatasari, T. Marhaeni, P. Astuti, and D. L. Setyowati, "Strategi Penanaman Nilai Kearifan Lokal Motif Sasirangan Dalam Keluarga Di Kampung Sasirangan Kota Banjarmasin," pp. 505–511, 2022.
- [5] T. T. Mukarromah *et al.*, "Gamifikasi Berbasis Aplikasi Dan Pembelajaran Anak Usia Dini," *J. Pertumbuhan, Perkembangan, dan Pendidik. Anak Usia Dini*, vol. 18, no. 229, pp. 18–27, 2021, doi: 10.17509/edukids.v18i1.33338.
- [6] H. Jusuf, L. S. Istiyowati, M. Fauzi, M. Magdalena, and R. Eko, "Metaverse-Based Learning in the Digital Era," vol. 25, no. December, pp. 334–346, 2023.
- [7] M. A. Permatasari, T. Marhaeni, P. Astuti, and D. L. Setyowati, "Strategies for Preserving Sasirangan and Planting Local Wisdom Values of Sasirangan Motifs in Schools in Banjarmasin , Indonesia," pp. 484–494, 2023.
- [8] C. Cinantya, A. Aslamiah, and A. Suriansyah, *Teacher Empowerment in Digitalization of Local Wisdom - Based Learning*, no. Icelet 2024. Atlantis Press SARL, 2025.
- [9] R. Sistem, "Identifikasi Level Pengelolaan Tata Kelola SIPERUMKIM Kota Salatiga berdasarkan COBIT 2019," *J. RESTI*, vol. 5, pp. 429–438, 2021, doi: 10.29207/resti.v5i3.3060.
- [10] et al Trisnadewi, "IT Governance Audit at District X Communications and Information Office Using COBIT 5," *JITE (Journal Informatics Telecommun. Eng.)*, vol. 5, pp. 360–370, 2022, doi: 10.31289/jite.v5i2.6183.
- [11] D. Anapaki, P. Alfa, and R. Leo, "Application of Pattern Recognition Method to Detect East Sumba Fabric Motifs in Lambanapu Using Convolutional Neural Network Method," *J. Artif. Intell. Eng. Appl.*, vol. 5, no. 1, pp. 2–9, 2025.
- [12] A. Y. Nugroho, S. H. Priyanto, T. Hendratono, H. Herryani, and F. Pradapa, "Digital transformation of the cultural heritage of the mataram kingdom," vol. 2, no. 3, pp. 3284–3294, 2025.
- [13] S. Xu, H. Chen, R. Ni, M. He, Z. Ge, and X. Liang, "Multi-Scale attention and hierarchical feature extraction-based network for metal surface defect detection," vol. 32, no. November, 2025.
- [14] S. Aulia and D. Rahmat, "Brain Tumor Identification Based on VGG-16 Architecture and CLAHE Method," *Int. J. INFORMATICS Vis.*, vol. 6, no. March, pp. 96–102, 2022.
- [15] H. Sastypratiwi and H. Muhardi, "Batik Recognition and Classification Using Transfer Learning and MobileNet Approach," vol. 8, no. December, pp. 2400–2410, 2024, doi: 10.62527/joiv.8.4.2407.
- [16] I. Faturrahman, M. Djamaluddin, Z. Amri, and M. N. Wathani, "Klasifikasi Motif Batik Nusantara Menggunakan Vision Transformer (ViT) Berbasis Deep Learning Imam," vol. 8, no. 2, pp. 511–522, 2025.
- [17] H. Wang, Z. Wang, X. Wang, Z. Wu, Y. Yuan, and Q. Li, "Neurocomputing AnatSwin: An anatomical structure-aware transformer network for cardiac MRI segmentation utilizing label images," *Neurocomputing*, vol. 577, no. February, p. 127379, 2024, doi: 10.1016/j.neucom.2024.127379.
- [18] N. A. Manap, L. X. Xuan, K. K. Singh, A. S. Akbari, and A. Putra, "Classification of Malaysian and Indonesian Batik Designs Using Deep Learning Models," *J. Telecommun. Electron. Comput. Eng.*, vol. 16, no. 4, pp. 23–30, 2024, doi: 10.54554/jtec.2024.16.04.004.
- [19] I. Khajenasiri, A. Estebarsari, M. Verhelst, and G. Gielen, "A review on Internet of Things solutions for intelligent energy control in buildings for smart city applications," *Energy Procedia*, vol. 111, no. September 2016, pp. 770–779, 2017, doi: 10.1016/j.egypro.2017.03.239.
- [20] R. Andrian, R. Taufik, D. Kurniawan, A. S. Nahri, and H. C. Herwanto, "Lampung Batik Classification Using AlexNet , EfficientNet , LeNet and MobileNet Architecture," *Int. J. Adv. Comput. Sci. Appl.*, vol. 15, no. 11, pp. 930–935, 2024.