Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK) DOI: 10.25126/jtiik.2025128932
Vol. 12, No. 1, Februari 2025, hlm. 77-86 p-ISSN: 2355-7699
Akreditasi KEMENRISTEKDIKTI, No. 36/E/KPT/2019 e-ISSN: 2528-6579

PENERAPAN TEACHABLE MACHINE DAN RASPBERRY PI PADA SISTEM KLASIFIKASI CITRA UNTUK INSPEKSI CACAT KAIN

Emmanuel Agung Nugroho[1*], Joga Dharma Setiawan[2], M. Munadi[3], Alifa Rustiyanti[4]

[1, 4]Politeknik Enjinering Indorama, Purwakarta, [1, 2, 3]Universitas Diponegoro, Semarang
Email:[1]emmanuel.agung@pei.ac.id, [2]joga.setiawan@gmail.com, [3]munadi@ft.undip.ac.id
[4]alifarustiyantii@gmail.com
[*]Penulis Korespondensi

(Naskah masuk: 16 April 2024, diterima untuk diterbitkan: 10 Februari 2025)

Abstrak

Industri tekstil memainkan peran krusial dalam ekonomi nasional, menghadapi tantangan signifikan dalam menjaga kualitas produk untuk memenuhi kepuasan konsumen. Cacat produksi, seperti cacat jarang dan cacat slap pada kain, merupakan salah satu faktor utama yang mempengaruhi kualitas produk tekstil. Penelitian ini bertujuan untuk mengembangkan sistem inspeksi cacat kain secara otomatis dengan menggunakan metode pemrosesan citra digital dan machine learning. Sistem ini dirancang untuk diintegrasikan pada mesin penggulungan kain sebagai sistem inspeksi awal sebelum kain didistribusikan. Metode yang digunakan meliputi supervised learning untuk klasifikasi citra kain, memanfaatkan perangkat lunak Google Teachable Machine dan algoritma Convolutional Neural Network (CNN) yang diimplementasikan dengan OpenCV. Perangkat keras yang digunakan terdiri dari kamera web Logitech D320 untuk akuisisi gambar dan Raspberry Pi-3B sebagai pengolah citra. Sistem ini diuji untuk mendeteksi tiga kategori kain: kain bagus, cacat jarang, dan cacat slap. Hasil pengujian menunjukkan bahwa sistem memiliki rata-rata waktu inferensi sebesar 142,47 ms dengan kecepatan rata-rata 6,46 frame per detik (FPS) dan akurasi klasifikasi mencapai 98,48%. Dengan implementasi sistem ini, diharapkan dapat meningkatkan efisiensi produksi, memperkuat kontrol kualitas di industri tekstil, mengurangi intervensi manual, dan menurunkan potensi kerugian akibat produk cacat.

Kata kunci: *Klasifikasi citra, Machine learning, Teachable Machine, Inspeksi cacat kain, Raspberry Pi*

APPLICATION OF TEACHABLE MACHINE AND RASPBERRY PI ON IMAGE CLASSIFICATION SYSTEM FOR FABRIC DEFECT INSPECTION

The textile industry plays a crucial role in the national economy, facing significant challenges in maintaining product quality to meet consumer satisfaction. Production defects, such as rare defects and slap defects in fabrics, are key factors that affect the quality of textile products. This research aims to develop an automated fabric defect inspection system using digital image processing and machine learning methods. The system is designed to be integrated into fabric winding machines as an initial inspection system before the fabric is distributed. The methods used include supervised learning for fabric image classification, utilizing Google Teachable Machine software and the Convolutional Neural Network (CNN) algorithm implemented with OpenCV. The hardware used consists of a Logitech D320 webcam for image acquisition and a Raspberry Pi-3B as the image processor. The system was tested to detect three categories of fabric: good fabric, rare defects, and slap defects. The test results showed that the system achieved an average inference time of 142.47 ms with an average speed of 6.46 frames per second (FPS) and a classification accuracy of 98.48%. With the implementation of this system, it is expected to enhance production efficiency, strengthen quality control in the textile industry, reduce manual intervention, and decrease potential losses due to defective products.

Keywords: *Image classification, Machine learning, Teachable Machine, Fabric defect inspection, Raspberry Pi.*

1. PENDAHULUAN

Dikutip dari Website resmi Kementerian Perindustrian Republik Indonesia pada hari Kamis, 14 Juli 2022, Menteri Perindustrian menyampaikan bahwa industri Tekstil dan Produk Tekstil (TPT) memiliki peran strategis dalam pembangunan nasional, dengan kontribusi signifikan dalam menyerap tenaga kerja sebesar 3,65 juta orang per Agustus 2021. Pertumbuhan pesat industri tekstil ini diiringi oleh tantangan besar, yaitu peningkatan daya saing kualitas produk untuk memenuhi kepuasan pelanggan. Menurut Crosby, kualitas adalah kesesuaian dengan kebutuhan, yang mencakup *availability, delivery, reliability, maintainability* dan

cost effectiveness (Kasus et al., n.d.). Cacat produksi merupakan salah satu faktor utama yang mempengaruhi kualitas suatu produk. Dalam produksi kain, beberapa cacat yang sering terjadi antara lain double lusi, double pick, ketebalan yang tidak merata, slap, jarang, warna kain yang tidak sesuai, dan corak yang meleset (Arif et al., 2018).

1.1 Tujuan Penelitian

Penelitian ini bertujuan untuk merealisasikan alat yang mampu mendeteksi cacat kain secara otomatis, yang berfungsi sebagai *early warning inspection* agar rekomendasinya dapat digunakan untuk menentukan faktor-faktor kualitas produk. Alat ini ditempatkan pada mesin penggulungan kain sebelum kain didistribusikan ke konsumen. Jenis cacat yang dideteksi dalam penelitian ini ada dua, yaitu cacat jarang yaitu adanya jarak antar benang pada kain yang lebih lebar dari yang lainnya, dan cacat slap yaitu adanya noda atau bintik- bintik hingga bergaris pada kain. Untuk menjalankan fungsinya sebagai pendeteksi cacat kain, dilakukan pemodelan kondisi kain menggunakan software Google Teachable Machine, dengan pengolahan citra untuk fungsi klasifikasi dijalankan oleh Raspberry Pi-3B (Ajhari et al., 2023). Google Teachable machine software adalah aplikasi open source dari Google yang mampu membangun aplikasi klasifikasi citra. Keluaran dari Teachable Machine adalah Tensorflow, tensorflow lite, atau data citra digital yang dikembangkan untuk menjalankan fungsi machine learning pada Raspberry Pi (Nugroho et al., 2024). Sistem yang digunakan dalam penelitian ini menggunakan metode *image classification*, yang merupakan salah satu bagian dari pemrosesan citra digital dan termasuk dalam rumpun *machine learning*. Dalam sistem klasifikasi citra ini, terdapat beberapa tahapan yang mencakup proses *supervised learning*, ekstraksi fitur, pengumpulan algoritma *machine learning*, peningkatan performa dengan data tambahan, serta adaptasi model melalui variasi data *training* (Id, 2021; Kampezidou et al., 2024).

Dengan diterapkannya teknologi ini dalam praktik industri, diharapkan dapat memberikan kontribusi signifikan dalam meningkatkan efisiensi proses inspeksi kain. Alat yang dikembangkan dalam penelitian ini mampu mendeteksi cacat secara otomatis dan lebih cepat dibandingkan dengan metode manual yang masih banyak diterapkan saat ini. Hal ini tidak hanya akan mengurangi biaya tenaga kerja tetapi juga mengurangi waktu inspeksi dan mengurangi potensi kesalahan manusia dalam mendeteksi cacat kain. Selain itu, sistem otomatis ini juga dapat memberikan umpan balik yang lebih cepat kepada produsen, sehingga dapat dilakukan penyesuaian pada tahap produksi untuk memastikan kualitas kain yang lebih konsisten sebelum didistribusikan ke konsumen. Peningkatan efisiensi dan kualitas ini secara langsung akan meningkatkan daya saing industri tekstil di pasar global, sekaligus memenuhi tuntutan pelanggan terhadap produk berkualitas tinggi.

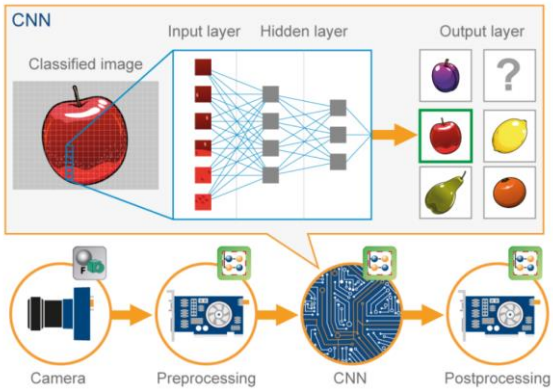
1.2 Teori Pendukung Penelitian

A. Machine learning

Machine Learning adalah salah satu cabang dari ilmu komputer dan kecerdasan buatan yang berfokus pada pengembangan sistem yang mampu "belajar" dari data tanpa harus diprogram secara eksplisit. Dalam konteks ini, belajar berarti bahwa sistem dapat menganalisis kumpulan data, mengenal pola, dan mengambil keputusan berdasarkan pola tersebut. Machine Learning secara langsung berkaitan dengan *data mining*, di mana komputer mampu mengoptimalkan kinerja mereka menggunakan sampel data yang tersedia (Nurhadi & Purnomo, 2022). Teknik teknik machine learning yang dapat diterapkan dalam sistem pengolahan citra diantaranya adalah:

1. Convolutional Neural Networks (CNNs): CNNs CNNs merupakan arsitektur jaringan saraf yang dirancang khusus untuk pengolahan gambar. CNN memanfaatkan lapisan konvolusi untuk mengekstrak fitur dari gambar, diikuti dengan lapisan pooling untuk mereduksi dimensi data tanpa kehilangan fitur penting. Teknik ini sangat kuat untuk tugas klasifikasi dan deteksi objek (Azmi et al., 2023; Kotta et al., 2022).
2. Transfer Learning: Transfer learning menggunakan model pra-latih, seperti model yang dilatih pada dataset besar seperti ImageNet, dan menyesuaikannya (fine-tuning) dengan dataset yang lebih kecil atau spesifik. Teknik ini menghemat waktu pelatihan dan meningkatkan performa model pada tugas-tugas tertentu dengan dataset yang terbatas. (Buyut Khoirul Umri & Delica, 2021).
3. Support Vector Machines: SVM adalah algoritma klasik dalam machine learning yang dapat digunakan untuk klasifikasi citra setelah fitur diekstraksi. Meskipun bukan deep learning, SVM cukup efektif untuk masalah klasifikasi berbasis fitur sederhana (Sanjaya & Ayub, 2020).
4. Random Forest: Random Forest adalah algoritma pembelajaran berbasis pohon keputusan yang dapat digunakan untuk klasifikasi gambar. Setelah fitur-fitur gambar diekstraksi, Random Forest bisa menghasilkan hasil yang stabil dan kuat, terutama dalam dataset yang bervariasi (Religia, 2019).

Dengan menggunakan kombinasi teknik-teknik ini, model machine learning dapat dibangun dan dilatih untuk melakukan aplikasi, seperti pengenalan objek, deteksi objek, dan klasifikasi objek (Handa et al., 2021). Gambar 1 adalah contoh algoritma aplikasi machine learning untuk klasifikasi citra.



Gambar 1 Ilustrasi algoritma CNN pada aplikasi image classification (Jaswal et al., 2014)

Pada Gambar 1 memberikan gambaran algoritma CNN untuk aplikasi image classification dilengkapi dengan peralatan yang diperlukan untuk proses klasifikasi citra seperti kamera, *preprocessing tool*, device untuk pengolahan algoritma CNN dan sarana untuk menampilkan hasil algoritma pada tahap post processing. Pada Penelitian ini objek yang dideteksi berupa jenis kain, preprocessing tool menggunakan *Google Teachable Machine*, algoritma CNN dan menampilkan output menggunakan Raspberry Pi-3B.

B. Image Classification

Klasifikasi citra *(image classification)* adalah proses untuk mengelompokkan citra digital ke dalam kategori yang telah ditentukan berdasarkan fitur visual yang terdeteksi dalam gambar tersebut. Tujuan utama dari klasifikasi citra adalah untuk mengidentifikasi atau memprediksi label yang sesuai dengan citra yang diberikan (Diantoro et al., 2019). Dalam pengolahan citra digital, dua pendekatan umum yang digunakan adalah supervised learning dan unsupervised learning (Boussemart et al., 2011). Supervised learning merupakan pendekatan di mana model dilatih menggunakan data yang sudah diberi label. Dataset pelatihan terdiri dari pasangan input dan output yang diketahui, yang memungkinkan model mempelajari hubungan antara input dan label yang ditetapkan (Cheng et al., 2014). Ada dua jenis utama dari supervised learning:

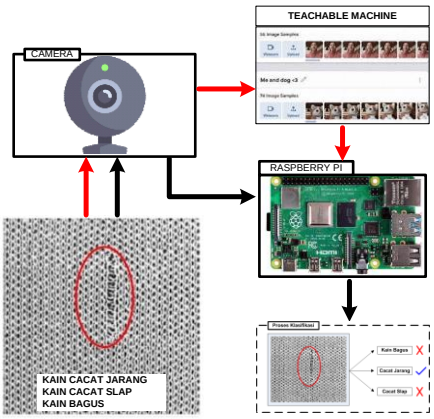
1. Regresi: Teknik ini digunakan untuk mempelajari hubungan antara input dan output kontinu, sehingga model dapat memprediksi output baru untuk input yang belum pernah dilihat sebelumnya (Lindholm et al., 2019), (Angelopoulos et al., 2022).
2. Klasifikasi: Teknik ini bertujuan untuk memetakan setiap input ke kelas atau label diskrit yang telah ditentukan. Contoh aplikasi klasifikasi meliputi pengenalan gambar, klasifikasi teks, dan deteksi spam pada email (Mishra et al., 2023).

Di sisi lain, unsupervised learning bekerja pada data yang tidak memiliki label atau informasi yang terstruktur. Tujuan utamanya adalah menemukan pola atau struktur tersembunyi dalam data tanpa bantuan label yang telah ditetapkan sebelumnya (Olaode et al., 2014). Dua tugas utama dalam unsupervised learning:

1. Clustering: Teknik ini bertujuan untuk memisahkan data ke dalam kelompok yang berbeda berdasarkan kesamaan fitur, tanpa adanya label. Contoh aplikasinya termasuk segmentasi pelanggan berdasarkan pola pembelian dan pengelompokan gambar berdasarkan kemiripan visual (Goldberger et al., 2006).
2. Dimensionality Reduction: Teknik ini berfungsi untuk mengurangi jumlah dimensi atau fitur dalam dataset sambil mempertahankan informasi yang relevan sebanyak mungkin. Metode ini sering digunakan untuk mempercepat proses pemodelan dan meningkatkan kinerja model pada data yang besar dan kompleks (Dash et al., 1997).

2. METODE PENELITIAN

Penelitian ini menggunakan metode supervised learning dengan data training berupa tiga kategori kain: kain bagus, kain dengan cacat jarang, dan kain dengan cacat slap. Dalam tahap pre-processing, alat bantu *Google Teachable Machine* digunakan untuk mengolah data citra awal. Algoritma *Convolutional Neural Network* (CNN) dikembangkan menggunakan OpenCV untuk melakukan klasifikasi citra kain. Dari sisi perangkat keras, kamera web Logitech D320 digunakan untuk menangkap citra objek, sementara Raspberry Pi-3B berperan dalam tahap post-processing, termasuk memanfaatkan GPIO untuk memberikan indikator dan peringatan berdasarkan hasil klasifikasi kain. Proses klasifikasi citra dalam penelitian ini diilustrasikan melalui Gambar 2.

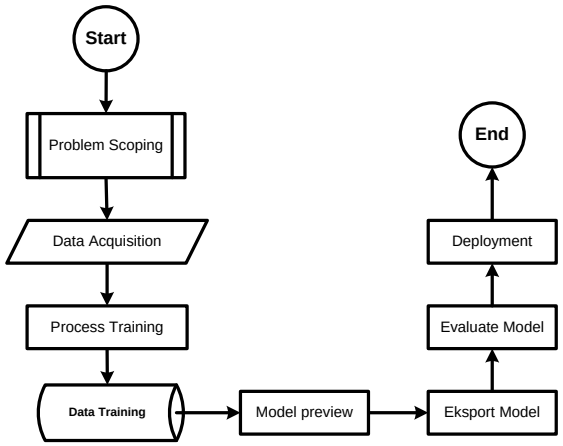


Gambar 2 Diagram proses sistem klasifikasi kain.

Gambar 2 mengilustrasikan alur proses sistem klasifikasi kain yang menggunakan Teachable Machine dan Raspberry Pi. Garis merah menunjukkan tahapan pelabelan dan pelatihan model, sementara garis hitam merepresentasikan proses identifikasi model. Diagram ini menegaskan bahwa sistem klasifikasi kain yang digunakan menerapkan metode supervised learning.

Dalam penelitian ini, data citra kain yang telah dikumpulkan diolah melalui Teachable Machine.

Data tersebut kemudian diubah menjadi format XML *(Extensible Markup Language)* sebelum dikonversi ke dalam kumpulan kode biner dan dikembalikan ke format gambar. Setiap gambar yang dihasilkan diberi label sesuai dengan kategori klasifikasinya. Proses lengkap dari algoritma yang digunakan dalam Teachable Machine dijelaskan lebih lanjut pada Gambar 3.



Gambar 3 Algoritma Sistem Deteksi Kain dengan Teachable Machine

Melalui Gambar 3 secara umum menjelaskan tahapan penggunaan teachable machine yaitu pengelompokan model kedalam kelas yang akan dipelajari oleh komputer, proses training untuk menghasilkan data latih, dan eksport model yang diakhiri dengan proses deployment model ke dalam GPIO.

2.1 Data Acquisition

Pada tahap ini, dilakukan pengumpulan dataset berupa gambar atau objek yang akan digunakan untuk melatih model klasifikasi. Setiap kategori objek dilabeli sesuai dengan kelasnya. Dalam penelitian ini, dataset terdiri dari citra kain yang dikategorikan menjadi tiga kelas: kain bagus**,** cacat jarang**,** dan cacat noda**.** Tabel 1 merangkum jumlah sampel data yang digunakan dalam proses pelatihan:

Tabel 1 Dataset Sampel Kain

NO	KATEGORI	JUMLAH SAMPEL
1	KAIN BAGUS	352
2	CACAT JARANG	365
3	CACAT SLAP	365

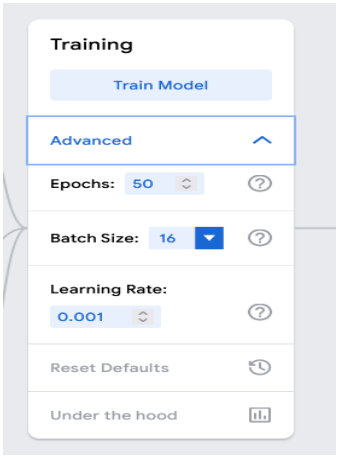
Tabel 1 menunjukkan bahwa terdapat tiga kategori sampel yang dikumpulkan. Proses pengumpulan data dilakukan menggunakan Teachable Machine dengan bantuan kamera Logitech D320, menghasilkan total 1.082 citra yang siap digunakan untuk melatih model klasifikasi dengan ukuran image rata-rata 9,5 kilobytes, dan ukuran original pixel dimension 147 K, dengan resolusi 28,246 pixels/cm.

2.2 Proses Training

Pada tahap ini, model machine learning dilatih menggunakan dataset yang telah dikumpulkan. Proses ini melibatkan penggunaan algoritma seperti *Convolutional Neural Networks (CNN)* untuk menyesuaikan parameter model sehingga dapat mengenali objek atau kategori yang diinginkan dengan akurasi tinggi. Beberapa istilah penting yang menjadi parameter dalam proses pelatihan meliputi:

- *Epochs*: Jumlah keseluruhan dataset yang digunakan dalam pelatihan model, dihitung berdasarkan berapa kali model mengakses seluruh dataset selama pelatihan. Parameter ini menentukan berapa banyak iterasi yang diperlukan hingga model mencapai hasil yang optimal (Afaq & Rao, 2020).
- *Batch Size*: Jumlah contoh data yang diproses dalam satu kali iterasi pelatihan. Batch size mempengaruhi kecepatan pelatihan serta penggunaan sumber daya komputasi selama proses berlangsung(Ali et al., 2020; Radiuk, 2018).
- *Learning Rate*: Parameter yang mengontrol besaran perubahan pada bobot model dalam setiap iterasi. Learning rate menentukan kecepatan konvergensi model menuju solusi optimal, jika nilai yang terlalu besar menyebabkan pelatihan tidak stabil, dan jika nilai yang terlalu kecil memperlambat proses pelatihan (Rochmawati et al., 2021).

Gambar 4 menunjukan proses training pada *Teachable Machine*



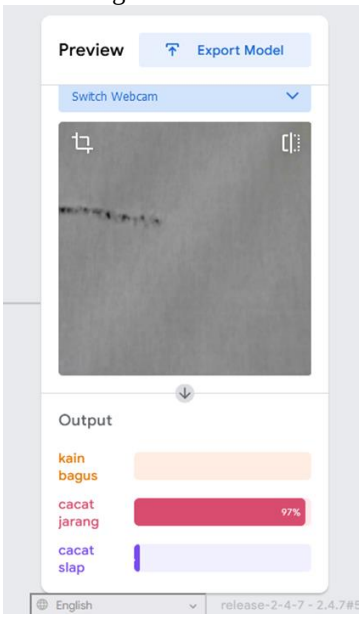
Gambar 4 Pelatihan Model Teachable Machine

Gambar 4 menampilkan proses training yang dilakukan pada pengambilan sampel cacat kain, terdapat 50 epoch, 16 batch size dan learning rate 0,001.

2.3. Tampilan Aplikasi

Pada tahap ini, dilakukan pengembangan antarmuka atau tampilan aplikasi yang memungkinkan integrasi model yang telah dilatih. Antarmuka ini dirancang agar pengguna dapat berinteraksi dengan model secara efektif. Dalam implementasi menggunakan *Teachable Machine*, tahap ini disebut sebagai Preview, yaitu proses di mana kesesuaian antara model yang dilatih dengan data training ditampilkan dalam bentuk persentase

akurasi. Gambar 5 menunjukan hasil preview model yang telah di training.

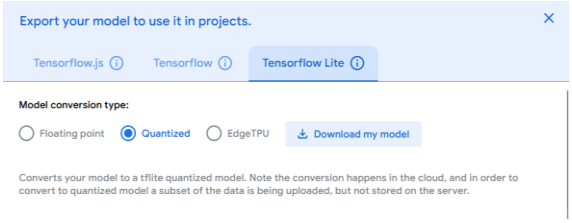


Gambar 5 Preview Model Teachable Machine

Pada Gambar 5**,** ditampilkan contoh proses preview di mana model mendeteksi cacat jarang dengan tingkat klasifikasi yang sangat tinggi, mencapai 97% akurasi. Hal ini menunjukkan bahwa model berhasil mengenali kategori cacat tersebut dengan baik berdasarkan data training yang diberikan.

2.4 Export

Langkah Pada langkah ini, model *Machine Learning* yang telah dilatih di *Teachable Machine* diekspor untuk digunakan dalam aplikasi lain. Proses ekspor ini menghasilkan model dalam format *TensorFlow Lite* dengan metode kuantisasi. TensorFlow Lite dengan kuantisasi adalah model yang dioptimalkan untuk berjalan pada perangkat dengan daya komputasi terbatas, seperti Raspberry Pi. Kuantisasi adalah teknik yang digunakan untuk mengurangi ukuran dan kompleksitas model dengan merepresentasikan angka-angka secara lebih efisien, sehingga memungkinkan model berjalan lebih cepat dan cocok untuk aplikasi real-time yang memerlukan respons cepat. Gambar 6 menunjukkan proses ekspor model pada Teachable Machine.



Gambar 6 Proses Export Model Teachable Machine

Setelah proses ekspor selesai, dua file dihasilkan:

1. *model.tflite*: berisi model yang telah dioptimalkan untuk penggunaan pada perangkat berdaya rendah.
2. *labels.txt*: berisi informasi mengenai kategori atau kelas yang telah didefinisikan dalam model.

2.5 Pengujian

Pengujian model bertujuan untuk memastikan bahwa model memberikan prediksi yang akurat dan aplikasi berfungsi dengan baik. Proses pengujian meliputi uji fungsionalitas serta verifikasi bahwa model menghasilkan hasil yang diharapkan ketika diterapkan pada data baru. Tahapan evaluasi dalam penelitian ini dilakukan untuk menilai kinerja model *Machine Learning* dengan menguji model pada data yang belum pernah dilihat sebelumnya. Evaluasi kinerja ini difokuskan pada penghitungan akurasi, yang dihitung dengan rumus berikut:

$$Akurasi = \left(\frac{Jumlah\ Percobaan\ benar}{Total\ Percobaan}\right) x\ 100\%$$

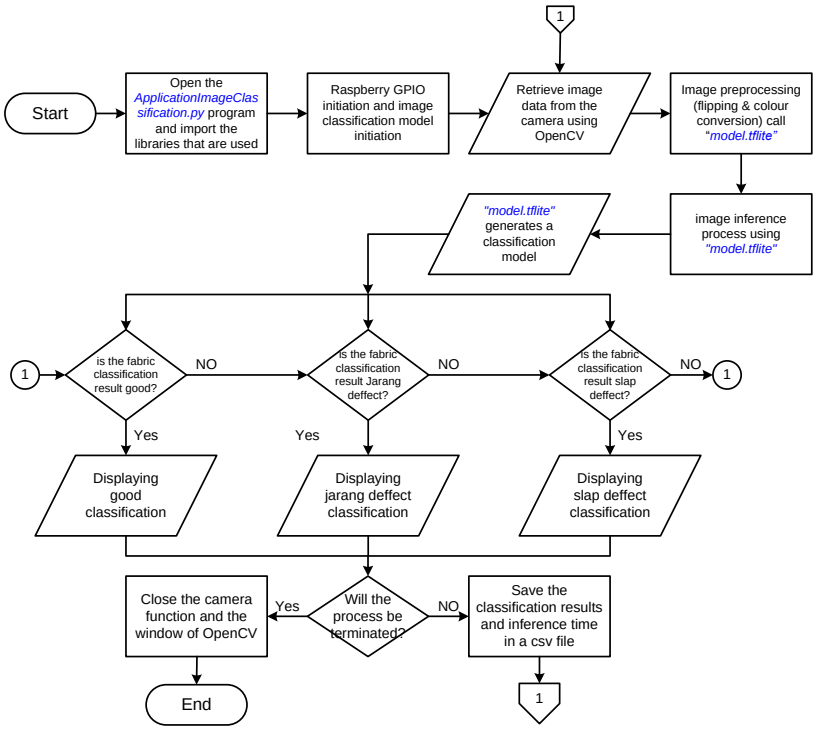
2.6 Deployment

Deployment adalah proses penanaman model TensorFlow ke dalam perangkat Raspberry Pi menggunakan *ApplicationImageClassification.py*. Untuk menjelaskan proses ini, algoritma program yang digunakan ditunjukkan pada Gambar 7**.**

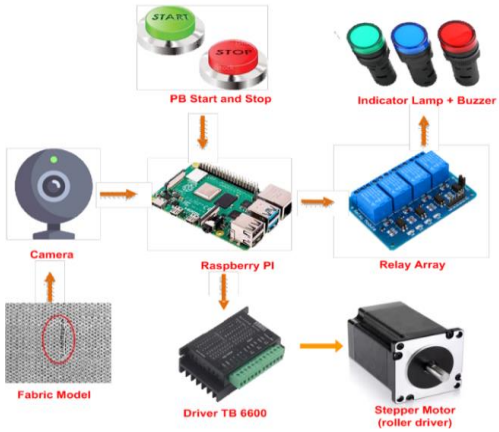
Gambar 7 menjelaskan secara rinci tahapan deployment program klasifikasi citra ke dalam Raspberry Pi menggunakan bahasa pemrograman Python. Proses ini dimulai dengan inisialisasi sistem dan konfigurasi pin GPIO pada Raspberry Pi**.** Selanjutnya, program mengambil gambar dari sumber kamera dan melakukan operasi pre-processing yang diperlukan, seperti flipping dan konversi warna. Gambar yang diperoleh dari tangkapan kamera dioperasikan dengan model klasifikasi citra yang telah diinisialisasi sebelumnya. Hasil inferensi dari model digunakan untuk melakukan klasifikasi terhadap gambar tersebut.

Jika hasil klasifikasi memenuhi kondisi yang telah ditentukan, program akan mengendalikan pin GPIO berdasarkan hasil klasifikasi tersebut. Proses ini berlanjut secara terus-menerus, kembali ke langkah pengambilan gambar dari kamera setelah menyelesaikan langkah-langkah klasifikasi dan pengendalian perangkat keras. Program akan berhenti dan menutup ketika pengguna menekan tombol *ESC*. Selanjutnya, Gambar 8 menggambarkan diagram blok sistem perangkat keras klasifikasi citra

Gambar 8 memberikan penjelasan mengenai proses komunikasi antar perangkat keras yang digunakan dalam aplikasi klasifikasi citra untuk mendeteksi cacat kain. Kamera mengambil sampel citra kain yang melintas dalam jangkauannya, kemudian gambar tersebut dikirim ke Raspberry Pi melalui port USB untuk diproses dengan algoritma CNN. Pengolahan data citra dilakukan menggunakan program *Application of Image Classification*, yang menghasilkan data klasifikasi kain.



Gambar 7 Algoritma pemrograman image classification dengan Raspberry Pi



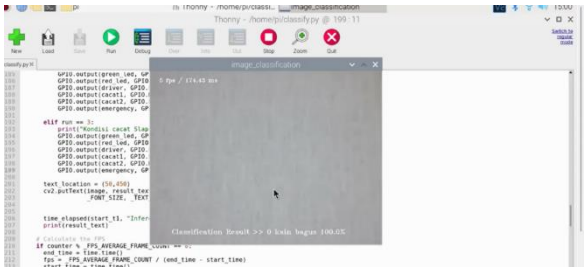
Gambar 8 Visualisasi hardware pada sistem klasifikasi citra

Berdasarkan hasil klasifikasi, program akan mengendalikan pin GPIO pada Raspberry Pi untuk menjalankan fungsi eksekusi program, seperti menghentikan rotasi motor penggulung, menyalakan indikator jenis cacat kain, dan mengaktifkan buzzer. Tombol *start* digunakan untuk mengaktifkan kembali proses penggulungan kain dengan memutar motor stepper dan menghidupkan kembali sistem klasifikasi. Sementara itu, tombol *stop* berfungsi untuk menghentikan sementara proses yang sedang berjalan.

3. HASIL DAN PEMBAHASAN

3.1 Hasil Application Image Classification

Dalam penelitian ini ada tiga variabel utama yang diamati yaitu data FPS *(Frame Per Second)*, waktu inference, dan akurasi untuk setiap kali proses deployment dengan aplikasi *ApplicationImageClassification.py*.



Gambar 9 Hasil aplikasi ApplicationImageClassification.py

Gambar 9 adalah tampilan dari *ApplicationImageClassification.py*. Dalam tampilan ini, terdapat beberapa informasi penting yang disajikan:

1. Data FPS *(Frames Per Second)*, menyatakan kecepatan perangkat kamera mampu menangkap frame citra dalam satu detik.
2. Data inference menyatakan waktu yang diperlukan untuk melakukan proses inferensi atau waktu yang dibutuhkan untuk melakukan prediksi atau klasifikasi terhadap data yang dibacanya.
3. Hasil klasifikasi adalah hasil pembacaan kamera yang menyatakan jenis klasifikasi objek model yang dibacanya.
4. Nilai klasifikasi adalah tingkat keyakinan atau tingkat akurasi sistem terhadap hasil klasifikasi yang diberikan.

3.2 Pengujian Performa Application Image Classification

Pengujian Performa Aplikasi klasifikasi citra mencakup pengukuran kecepatan proses klasifikasi dalam parameter *Inference Times*, tingkat kecepatan kamera membaca setiap frame satuan frame per detik

(FPS), dan tingkat akurasi pembacaan hasil klasifikasi. Nilai parameter inference times dan fps untuk setiap kategori ditunjukkan pada Tabel 2.

Tabel 2. Data pengujian kecepatan proses klasifikasi

Kategori	Rata-rata FPS	Rata-rata Inference (ms)	Jumlah Frame
Kain Bagus	7,45	126,31	350
Cacat Jarang	6,00	150,1	350
Cacat Slap	5,94	152,22	350

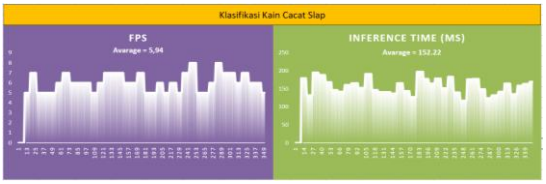
Berdasar Tabel 2 menyatakan kecepatan FPS dan kecepatan inferensi proses klasifikasi terhadap 3 jenis model yaitu kain bagus, kain cacat jarang dan kain cacat slap. Berdasar Tabel 2 juga dapat dibuat grafik performa *ApplicationImageClassification.py*.

Gambar 10 Grafik Hasil Performa Application Image Classification Kain Bagus

Gambar 10 menunjukkan hasil *frame per second* untuk klasifikasi kain bagus dengan hasil FPS 7,45 dan waktu inferensi 126,31.

Gambar 11 Grafik Hasil Performa Application Image Classification Cacat Jarang

Gambar 11 menunjukkan hasil *frame per second* untuk klasifikasi kain bagus dengan hasil FPS 6,0 dan waktu inferensi 150,1.



Gambar 12 Grafik Hasil Performa Application Image Classification Cacat Slap

Gambar 12 menunjukkan hasil *frame per second* untuk klasifikasi kain bagus dengan hasil FPS 5,94 dan waktu inferensi 122,22.

Dari tiga hasil tersebut, dapat diambil kesimpulan bahwa *Application Image Classification* ini dapat menghasilkan klasifikasi dengan waktu inferensi yaitu dengan rata rata pada semua kategori sebesar 142,47 ms dan rata rata FPS pada semua kategori sebesar 6,46 FPS.

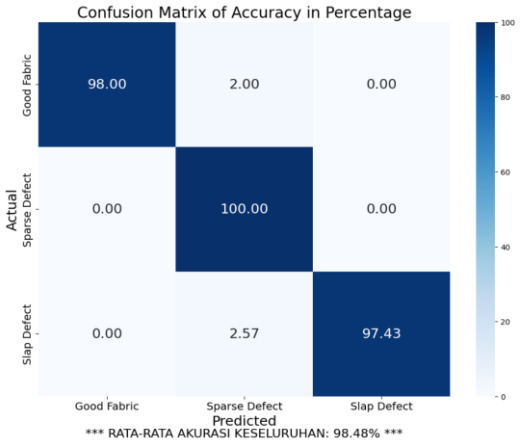
3.3 Pengujian Akurasi Klasifikasi Model

Pengujian Akurasi untuk mengevaluasi seberapa baik model atau sistem tersebut dalam mengklasifikasikan citra-citra ke dalam kelas atau label yang tepat. Data hasil pengujian akurasi diperoleh dari file 'hasil_klasifikasi.csv'. Tabel 3 menunjukan data prediksi keakuratan sistem yang telah diambil dari file csv.

Tabel 1 Data Hasil akurasi sistem Klasifikasi kain

Prediksi	Waktu Inferensi	Aktual		
		Kain Bagus	Cacat Jarang	Cacat Slap
Kain Bagus	126,21	343	7	0
Cacat Jarang	150,1	0	350	9
Cacat Slap	152,22	0	9	341

Berdasarkan Tabel 3 menjelaskan bahwa pada pengujian kain bagus yaitu selama 126,69 ms terbaca 350 hasil inferensi dengan 343 menyatakan sebagai kain bagus dan 7 menyatakan sebagai cacat jarang. Pada pengujian cacat jarang yaitu selama 150,87 ms terbaca 350 hasil inferensi dengan 350 menyatakan sebagai cacat jarang. Pada pengujian cacat slap yaitu selama 152,93 ms terbaca 350 hasil inferensi dengan 341 menyatakan sebagai cacat slap dan 9 menyatakan sebagai cacat jarang. Visualisasi *confusion matrix* dalam bentuk persentase berdasarkan data yang terdapat pada Tabel 3 ditunjukkan pada Gambar 13



Gambar 13 Confusion Matrix Akurasi Klasifikasi Kain

Berdasarkan hasil pada Gambar 13, diperoleh hasil rata-rata akurasi dari seluruh hasil pengujian klasifikasi model sebesar 98,48%.

3. 4. KESIMPULAN

Sistem klasifikasi citra untuk aplikasi inspeksi cacat kain dengan menggunakan tools Teachable Machine dan Raspberry Pi-3 pada penelitian ini menunjukkan keterkaitan antara waktu inferensi dengan FPS untuk setiap kategori kondisi benang. Dalam 3 kali pengujian diperoleh data semakin tinggi waktu inferensi maka semakin rendah FPS. Pembacaan hasil tingkat akurasi sangat dipengaruhi

oleh sistem pencahayaan pada permukaan objek kain yang di inspeksi. Pada tingkat kecerahan optimal diperoleh akurasi hingga 98,48% pada pembacaan confusion matrix. Untuk penelitian di masa depan, disarankan untuk mengeksplorasi penggunaan perangkat keras yang lebih canggih atau kombinasi dengan algoritma pengolahan citra yang lebih kompleks untuk meningkatkan ketahanan sistem terhadap perubahan kondisi pencahayaan. Penelitian lebih lanjut juga bisa mempertimbangkan penggunaan dataset yang lebih besar dan beragam untuk meningkatkan kemampuan generalisasi model.

DAFTAR PUSTAKA

AFAQ, S., & RAO, S. 2020. Significance Of Epochs On Training A Neural Network. *International Journal of Scientific & Technology Research*, *9*(06), 1–4. www.ijstr.org

AJHARI, A. A., SITORUS, K. J., LYRAWATI, D. P. N., SRIDANA, E. F. S., LESTARI, R., & FARIDA, W. V. 2023. Noaa Image Classification Using Google'S Teachable Machine. *Semnas Ristek (Seminar Nasional Riset Dan Inovasi Teknologi)*, *7*(1), 43–49. https://doi.org/10. 30998/semnasristek.v7i1.6244

ALI, A., PINCIROLI, R., YAN, F., & SMIRNI, E. 2020. Batch: Machine learning inference serving on serverless platforms with adaptive batching. *International Conference for High Performance Computing, Networking, Storage and Analysis, SC*, *2020-Novem*. https://doi.org/10.1109/ SC41405.2020.00073

ANGELOPOULOS, A. N., KOHLI, A., BATES, S., JORDAN, M. I., MALIK, J., ALSHAABI, T., UPADHYAYULA, S., & ROMANO, Y. 2022. Image-to-Image Regression with Distribution-Free Uncertainty Quantification and Applications in Imaging. *Proceedings of Machine Learning Research*, *162*, 717–730.

ARIF, M. S., PUTRI, C. F., & TJAHJONO, N. 2018. *Peningkatan Grade Kain Sarung dengan Mengurangi Cacat Menggunakan Metode Kaizen dan Siklus PDCA pada PT . X*. *26*(2), 222–231.

AZMI, K., DEFIT, S., & SUMIJAN, S. 2023. Implementasi Convolutional Neural Network (CNN) Untuk Klasifikasi Batik Tanah Liat Sumatera Barat. *Jurnal Unitek*, *16*(1), 28–40. https://doi.org/10.52072/unitek.v16i1.504

BOUSSEMART, Y., CUMMINGS, M. L., LAS FARGEAS, J., & ROY, N. 2011. Supervised vs unsupervised learning for operator state modeling in unmanned vehicle settings. *Journal of Aerospace Computing, Information and Communication*, *8*(3), 71–85. https://doi.org/ 10.2514/1.46767

BUYUT KHOIRUL UMRI, & DELICA, V. 2021. Penerapan transfer learning pada convolutional neural networks dalam deteksi covid-19. *Jnanaloka*, 9–17. https://doi.org/10.36802/jnanaloka.2021 .v2-no2-9-17

CHENG, P., CHIEN, B., & YANG, W. 2014. *Medical Image Classification by Supervised Machine Learning*. *June*.

DASH, M., LIU, H., & YAO, J. 1997. Dimensionality reduction of unsupervised data. *Proceedings of the International Conference on Tools with Artificial Intelligence*, *April*, 532–539. https://doi.org/10.1109/tai.1997.632300

DIANTORO, K., ADRIASYAH, B., INTEGRAL, C., & AWAL, P. C. 2019. *Sistem Identifikasi Jenis Burung Dengan Image*. *20*(1), 96–105.

GOLDBERGER, J., GORDON, S., & GREENSPAN, H. 2006. Unsupervised image-set clustering using an information theoretic framework. *IEEE Transactions on Image Processing*, *15*(2), 449–458. https://doi.org/10.1109/TIP.2005.860593

HANDA, N., KAUSHIK, Y., SHARMA, N., DIXIT, M., & GARG, M. 2021. Image Classification Using Convolutional Neural Networks. *Communications in Computer and Information Science*, *1393* (December 2022), 510–517. https://doi.org/10.1007/978-981-16-3660-8_48

ID, I. D. 2021. *Machine Learning : Teori , Studi Kasus dan Implementasi Menggunakan Python*. *July*. https://doi.org/10.5281/zenodo. 5113507

JASWAL, D., V, S., & SOMAN, K. P. 2014. Image Classification Using Convolutional Neural Networks. *International Journal of Scientific and Engineering Research*, *5*(6), 1661–1668. https://doi.org/10.14299/ijser.2014.06.002

KAMPEZIDOU, S. I., RAY, A. T., BHAT, A. P., FISCHER, O. J. P., & MAVRIS, D. N. 2024. *Fundamental Components and Principles of Supervised Machine Learning Workflows with Numerical and Categorical Data*. 384–416.

KASUS, S., ISKANDAR, P. T., PRINTING, I., DEWANTI, D. F., & PUJOTOMO, D. (n.d.). *Analisis Penyebab Cacat Produk Kain Dengan Menggunakan Metode Failure Mode And Effect Analysis (FMEA)*.

KOTTA, C. R., PASERU, D., & SUMAMPOUW, M. 2022. Implementasi Metode Convolutional Neural Network untuk Mendeteksi Penyakit Pada Citra Daun Tomat. *Jurnal Pekommas*, *7*(2), 123–132. https://doi.org/10.56873/jpkm.v7i2.4961

LINDHOLM, A., WAHLSTRÖM, N., LINDSTEN, F., & SCHÖN, T. B. 2019. *Supervised Machine Learning: Statistical Machine Learning course*. 112. http://www.it.uu.se/edu/course/homepage/sml/literature/lecture_notes.pdf

MISHRA, D. P., MISHRA, S., JENA, S., & SALKUTI, S. R. 2023. Image classification using machine learning. *Indonesian Journal of Electrical Engineering and Computer Science*,

31(3), 1551–1558. https://doi.org/10.11591/ijeecs. v31.i3.pp1551-1558

NUGROHO, E. A., SETIAWAN, J. D., MUNADI, & DIKI. 2024. Design of image classification system for fabric inspection process using Raspberry Pi. *Journal of Mechatronics, Electrical Power, and Vehicular Technology*, *15*(1), 57–67. https://doi.org/10.55981/j.mev.2024.863

NURHADI, M., & PURNOMO, J. 2022. Implementation of Image Classification Using Convolutional Neural Network (Cnn) Algorithm on Vehicles Images. *ASEAN Journal of Systems Engineering*, *6*(1), 1–5. https://doi.org/10.22146/ajse.v6i1.72411

OLAODE, A., NAGHDY, G., & TODD, C. 2014. Unsupervised classification of images: a review. *International Journal of Image Processing.*, *8.5*, 325–342. https://www.researchgate.net/profile/Abass-Olaode/publication/265729668_Unsupervised_Classification_of_Images_A_Review/links/541a74be0cf203f155ae295a/Unsupervised-Classification-of-Images-A-Review.pdf

RADIUK, P. M. 2018. Impact of Training Set Batch Size on the Performance of Convolutional Neural Networks for Diverse Datasets. *Information Technology and Management Science*, *20*(1), 20–24. https://doi.org/10.1515/itms-2017-0003

RELIGIA, Y. 2019. Feature Extraction Untuk Klasifikasi Pengenalan Wajah Menggunakan Support Vector Machine Dan K-Nearest Neighbor. *Pelita Teknologi: Jurnal Ilmiah Informatika, Arsitektur Dan Lingkungan*, *14*(2), 85–92.

ROCHMAWATI, N., HIDAYATI, H. B., YAMASARI, Y., TJAHYANINGTIJAS, H. P. A., YUSTANTI, W., & PRIHANTO, A. 2021. Analisa Learning Rate dan Batch Size pada Klasifikasi Covid Menggunakan Deep Learning dengan Optimizer Adam. *Journal of Information Engineering and Educational Technology*, *5*(2), 44–48. https://doi.org/10.26740/ jieet.v5n2.p44-48

SANJAYA, J., & AYUB, M. 2020. Augmentasi Data Pengenalan Citra Mobil Menggunakan Pendekatan Random Crop, Rotate, dan Mixup. *Jurnal Teknik Informatika Dan Sistem Informasi*, *6*(2), 311–323. https://doi.org/10.28932/jutisi.v6i2.2688

Halaman ini sengaja dikosongkan