

Sistem Klasifikasi Citra untuk Proses Inspeksi Kain Menggunakan Teachable Machine dan Raspberry Pi

Image Classification System for Fabric Inspection Using Teachable Machine and Raspberry Pi

Diki¹, Emmanuel Agung Nugroho², Nanang Roni Wibowo³

^{1,2,3} Mechatronics Engineering Department, Indorama Engineering Polytechnic, Purwakarta, Indonesia

¹ mulyadidiki29@gmail.com , ² emmanuel.agung@pei.ac.id , ³ nanang.roni@pei.ac.id

*Corresponding author: emmanuel.agung@pei.ac.id

Abstrak. Industri tekstil telah berkembang sangat pesat, bahkan pertumbuhan produk tekstil melebihi pertumbuhan ekonomi nasional. Permintaan produk tekstil tidak hanya untuk konsumsi dalam negeri tetapi juga untuk ekspor. Dalam upaya memenuhi standar kualitas dan menjaga kepuasan pelanggan, pengendalian kualitas produksi kain menjadi sangat penting, terutama untuk mengendalikan cacat produksi kain. Jenis-jenis cacat yang ada pada kain adalah lubang, noda, cacat langka karena benang putus/hilang, mengambang, luntur warna, pecah pola, benang ganda, benang tebal (slub), mixed end, pin mark, dan lain-lain. Pada penelitian ini, dirancang sebuah sistem yang dapat mendeteksi kecacatan produksi pada kain dengan menggunakan metode pengolahan citra berbasis machine learning menggunakan Raspberry Pi. Jenis cacat yang dimodelkan adalah cacat jarang dan cacat noda, atau dalam istilah pabrik sering disebut cacat slap. Hasil pengujian menunjukkan bahwa sistem ini memiliki rata-rata frame per second (FPS) sebesar 4.85, rata-rata waktu inferensi sebesar 181.1 ms, dengan akurasi hasil klasifikasi citra sebesar 98.4%.

Kata kunci: *machine learning; image Processing; Raspberry PI; inference time; image classification.*

Abstract. The textile industry has been growing rapidly, even the growth of textile products exceeds the growth of the national economy. The demand for textile products is not only for domestic consumption but also for export. In an effort to meet quality standards and maintain customer satisfaction, quality control of fabric production is very important, especially in controlling fabric production defects. The types of defects that exist in fabrics are holes, stains, rare defects due to broken/lost threads, floating, color fading, broken patterns, double threads, thick threads (slubs), mixed ends, pin marks, and others. In this research, a system is designed that can detect production defects in fabrics using machine learning-based image processing methods using Raspberry Pi. The types of defects modeled are sparse defects and stain defects, or in factory terms often called slap defects. The test results show that this system has an average frame per second (FPS) of 4.85, an average inference time of 181.1 ms, with an image classification result accuracy of 98.4%.

Keywords: machine learning; image Processing; Raspberry PI; inference time; image classification.

1 Pendahuluan

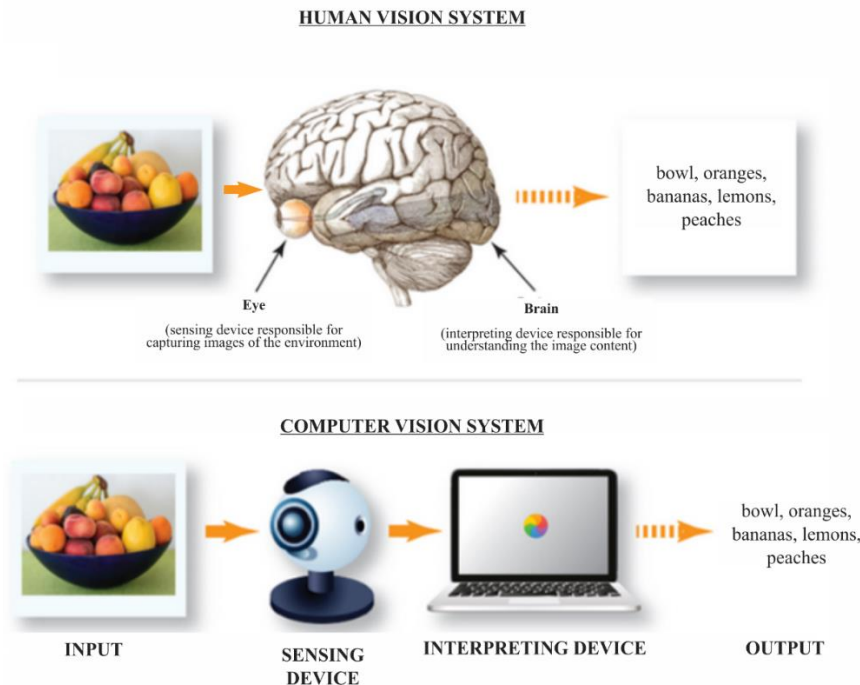
Industri tekstil dan garmen merupakan sektor manufaktur yang mencatatkan pertumbuhan tertinggi pada triwulan III tahun 2019, yaitu sebesar 15,35% (Kementerian Perindustrian, 2021). Industri tekstil merupakan salah satu dari lima sektor manufaktur yang diprioritaskan pengembangannya, terutama dalam rangka persiapan menuju Era Industri 4.0, berdasarkan peta jalan Making Indonesia (Kementerian Perindustrian, 2019). Perusahaan tekstil terus berupaya untuk menjaga kualitas produk mereka untuk memastikan kepuasan pelanggan. Cacat pada suatu produk tidak hanya mempengaruhi tingkat kepuasan pelanggan. Hal ini juga mempengaruhi efisiensi produksi perusahaan (Dewanti et al., 2017; Kristanto et al., 2016).

Tujuan dari penelitian ini adalah untuk membangun sebuah alat untuk mendeteksi kecacatan pada produksi kain dengan cara menangkap gambar kecacatan kain menggunakan kamera yang terhubung dengan alat aplikasi Teachable Machine (Chazar & Rafsanjani, 2022; Malahina et al., 2022; Pratama et al., 2022). Teachable Machine merupakan aplikasi open source dari Google yang mampu membangun aplikasi klasifikasi citra. Keluaran dari Teachable Machine adalah Tensorflow atau data citra digital yang dikembangkan untuk menjalankan fungsi machine learning pada Raspberry Pi (Maulana, 2021; Setiawan et al., 2022; Wijaya et al., 2017).

2 Kajian Pustaka

2.1 Computer Vision

Computer vision adalah bagian dari Deep Learning yang memungkinkan komputer untuk memproses dan menganalisis data visual dalam bentuk gambar atau video dengan cara meniru kecerdasan manusia. Tujuan utama dari Computer vision adalah memungkinkan komputer untuk melihat, menafsirkan, dan merepresentasikan dunia di sekitarnya secara visual. Gambar 1 menjelaskan bagaimana sistem penglihatan manusia berbeda dengan sistem penglihatan komputer (Alaskar & Saba, 2021; Bandyopadhyay, 2022; Han et al., 2018; Prasad et al., 2022; Purno & Wibowo, 2016)



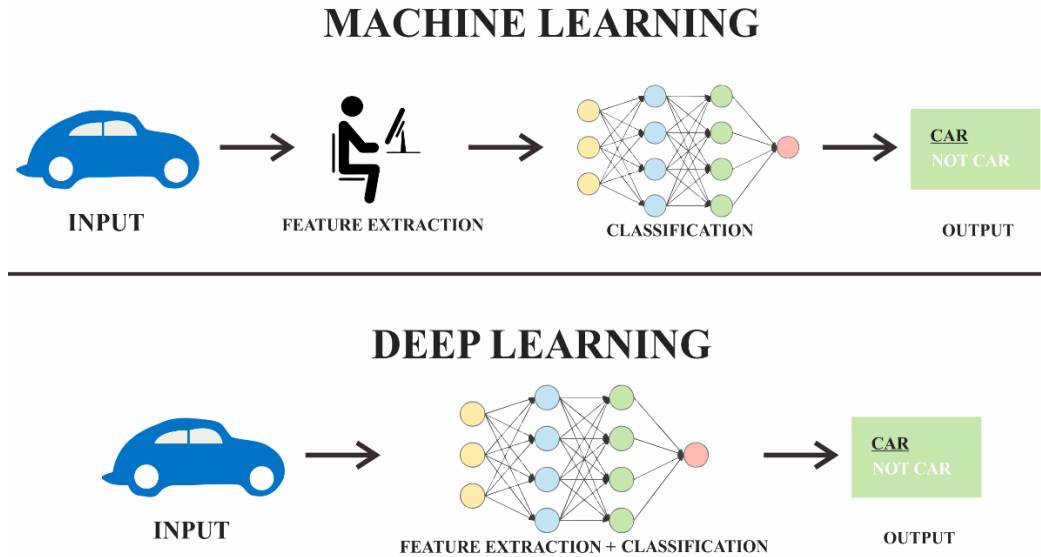
Gambar 1. Human Vision vs Computer Vision(Bandyopadhyay, 2022)

Computer vision adalah teknologi computer untuk aplikasi pemrosesan gambar, Machine Learning, jaringan saraf, dan algoritme visi robot. Data input berupa visual, seperti gambar dan video dibaca menggunakan kamera webcam dan komputer mengekstrak informasi dari data tersebut untuk menghasilkan pemahaman dalam gambar atau video. Computer vision memiliki berbagai aplikasi, termasuk otomasi industri, kendaraan otonom, pengawasan keamanan, pengenalan wajah, pemrosesan medis, pemrosesan gambar, dan banyak lagi. Teknologi ini terus berkembang dan menjadi semakin penting dalam dunia teknologi informasi modern (Dompeipen & Sompie, 2020)

2.2 Image Classification

Image classification adalah bagian dari Machine Learning, di mana kamera dan perangkat kontrol dioptimalkan untuk mengidentifikasi objek menggunakan teknik pemrosesan gambar dan data. Untuk mencapai klasifikasi yang benar, komputer dilatih untuk mengenali dan menginterpretasikan model berdasarkan sampel yang tersimpan dalam kumpulan data. Ada beberapa tahapan dalam proses image classification, yaitu training data, ekstraksi fitur, dan training model. Dalam implementasi machine learning, data training berupa sampel gambar dengan label tertentu yang

berfungsi sebagai model untuk mengidentifikasi karakter gambar. Ekstraksi fitur adalah proses mengidentifikasi, memilih, dan mengekstraksi informasi penting atau fitur yang relevan dari data training. Model Training adalah algoritma Machine Learning yang dilatih pada data training untuk menyimpulkan objek berdasarkan label kelas. Proses klasifikasi citra dalam kaitannya dengan Machine Learning dapat dijelaskan dengan menggunakan Gambar 2 (Abou Baker et al., 2022; Abu et al., 2019; Kamavisdar, P., Saluja, S., & Agrawal, 2013; Pintelas et al., 2020).

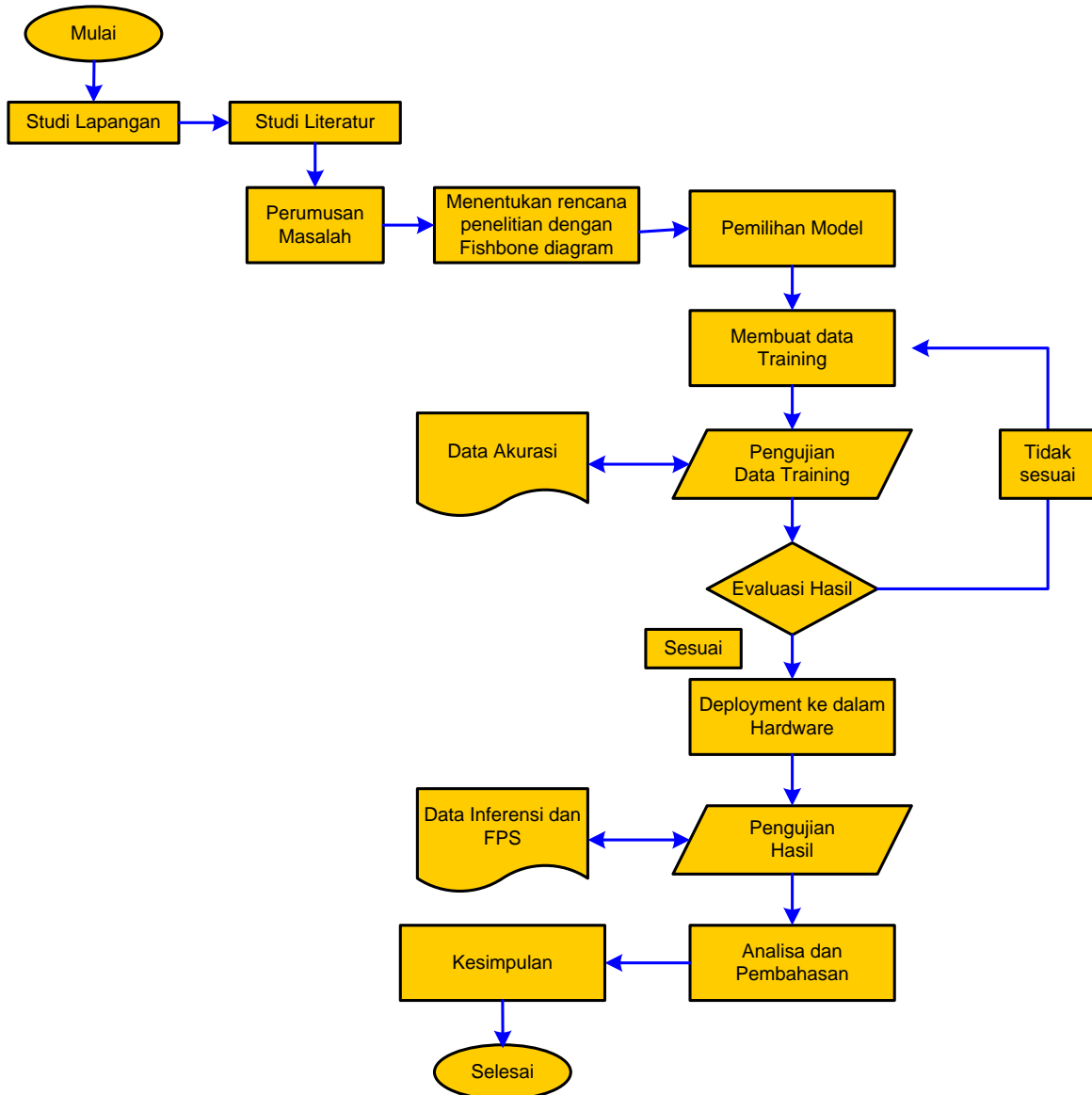


Gambar 2. *Machine Learning vs Deep Learning* (Alaskar & Saba, 2021)

Berdasarkan Gambar 2, dapat dijelaskan tahap image classification pada machine learning dan deep learning. Perbedaan proses klasifikasi citra antara machine learning dan deep learning terletak pada ekstraksi fitur. Pada machine learning, ekstraksi fitur dilakukan secara manual dengan bantuan manusia yang memiliki pengetahuan tentang domain data training. Pada deep learning, jaringan syaraf tiruan memiliki kemampuan untuk secara otomatis mengekstrak fitur yang relevan dari data training (Alaskar & Saba, 2021).

3 Metode Penelitian

3.1 Diagram Alir penelitian

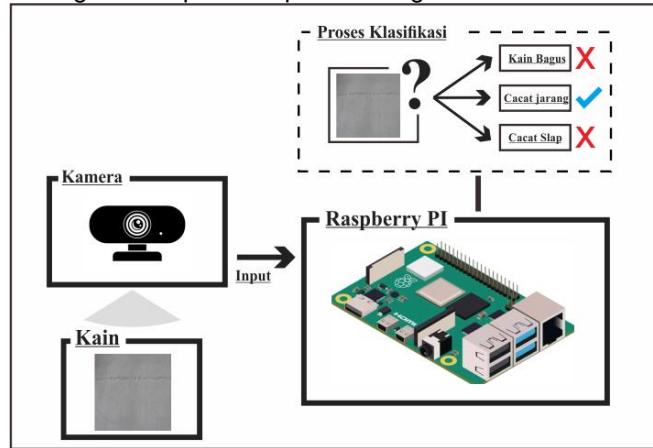


Gambar 3. Diagram Alir metode penelitian

Penelitian dimulai dengan survey lapangan di perusahaan xxx untuk mendapatkan informasi primer jenis-jenis cacat yang terjadi pada proses produksi kain. Dari informasi primer ini kemudian dilakukan kajian literature untuk mendapatkan data-data sekunder dari penelitian-penelitian yang sudah ada sebelumnya. Tahapan selanjutnya adalah melakukan perencanaan penelitian melalui fishbone diagram untuk mengidentifikasi obyek yang akan diteliti, software yang digunakan, hardware yang digunakan, sistem kontrol yang digunakan, dan metode pemrograman yang digunakan. Setelah obyek penelitian ditetapkan maka dilakukan pembuatan data training dengan menggunakan tools *Teachable Machine* dan mengujinya untuk mendapatkan akurasi yang paling optimal. Bila akurasi sudah sesuai dengan kriteria maka tahap selanjutnya adalah implementasi data training kedalam Hardware Raspberry Pi dan melakukan pengujian performa kerja Raspberry berdasarkan data inferensi dan frame per second yang dihasilkan. Dari data Akurasi dan data performa yang dihasilkan dibuat analisa untuk merumuskan kesimpulan keseluruhan pelaksanaan penelitian ini.

4 Hasil dan Pembahasan

Untuk merealisasikan sistem image classification pada penelitian ini, perangkat lunak Teachable Machine digunakan pada sistem input kamera untuk menghasilkan model training. Perangkat keras yang digunakan adalah kamera web Logitech D320 dan Raspberry Pi-4 untuk memproses model training dan mengalamatkan model klasifikasi ke GPIO untuk fungsi eksekusi pada bagian mekanik. Gambar 4 menunjukkan bagaimana proses aplikasi image classification bekerja.

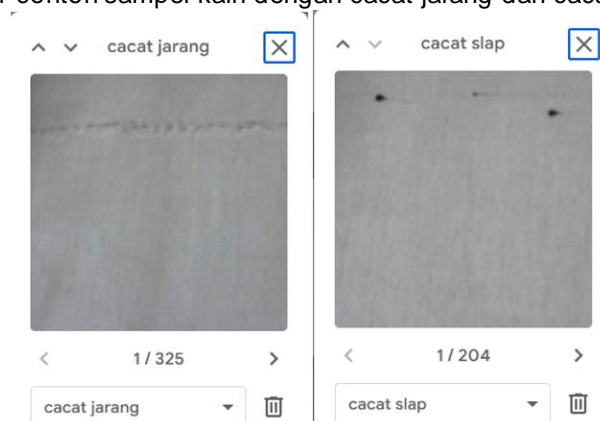


Gambar 4. Diagram Proses Image Classification untuk pendeteksi cacat kain

Dari Gambar 4, dapat dijelaskan bahwa kamera mengambil gambar objek berupa kain yang berada dalam jangkauannya, kemudian gambar tersebut dikirim ke Raspberry Pi melalui port USB untuk diproses lebih lanjut. Data citra tersebut diolah oleh program "Aplikasi Klasifikasi Citra" untuk mendapatkan hasil klasifikasi. Berdasarkan hasil klasifikasi yang diperoleh, program akan mengontrol pin GPIO Raspberry Pi untuk memberikan indikator dan instruksi kerja mekanik. GPIO Raspberry Pi juga mengatur relay dan LED sesuai dengan kondisi yang terdeteksi pada hasil klasifikasi. Secara spesifik, metodologi klasifikasi citra berdasarkan diagram proses pada Gambar 3 terdiri dari: Problem Scoping, Data Acquisition, Modelling, Extraction and Deployment.

4.1 Problem Scoping

Masalah yang telah diselidiki adalah jenis cacat jarang dan cacat slap. Cacat jarang adalah cacat karena benang yang putus/hilang pada lungsin. Cacat slap adalah kotoran atau noda pada kain. Gambar 5 menunjukkan contoh sampel kain dengan cacat jarang dan cacat noda atau slap.



Gambar 5. Model kain cacat jarang dan cacat slap

Gambar 5 adalah tampilan model pada alat Teachable Machine. Berdasarkan Gambar 5, model cacat jarang dan cacat slap digunakan sebagai contoh untuk membangun model image classification untuk sistem inspeksi kain. Dalam hal ini, dibutuhkan dua komponen utama, yaitu model dan aplikasi image classification. Dalam hal ini, model berperan sebagai "otak" atau inti yang melakukan pemrosesan dan pengambilan keputusan. Sedangkan aplikasi Image Classification berperan sebagai antarmuka yang memungkinkan pengguna atau sistem lain untuk berkomunikasi dengan model.

4.2 Data Acquisition

Tahap Akuisisi Data masih dalam aplikasi Teachable Machine, tahap ini berfungsi untuk mengumpulkan dataset dan memastikan kualitas dataset. Gambar 6 menunjukkan dataset yang diperoleh dalam proses klasifikasi model kain.



Gambar 6. Pengumpulan sampel dataset

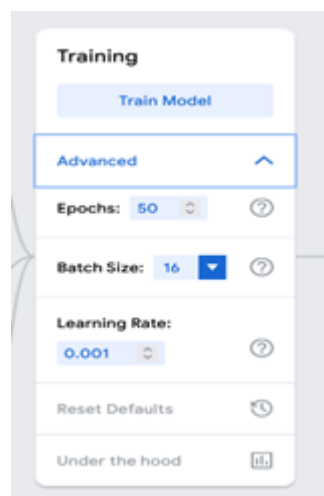
Gambar 6 menjelaskan tahap pengumpulan data training kain dengan mengklasifikasikannya ke dalam 3 kategori kelas, yaitu kain yang baik, cacat jarang dan cacat slap. Setiap data training memiliki sejumlah sampel untuk mengakomodasi berbagai jenis model untuk klasifikasi yang lebih akurat. Tabel 1 menunjukkan jumlah kategori dan jumlah sampel model kain.

Tabel 1 Sampel Dataset

NO	KATEGORI	JUMLAH SAMPEL
1	KAIN BAGUS	251
2	CACAT JARANNG	325
3	CACAT SLAP	204

4.3 Modelling

Modelling dicapai melalui proses training pada alat Teachable Machine. Modeling menentukan seberapa cepat atau lambat model konvergen ke solusi optimal. Gambar 6 menunjukkan proses training pada Teachable Machine.



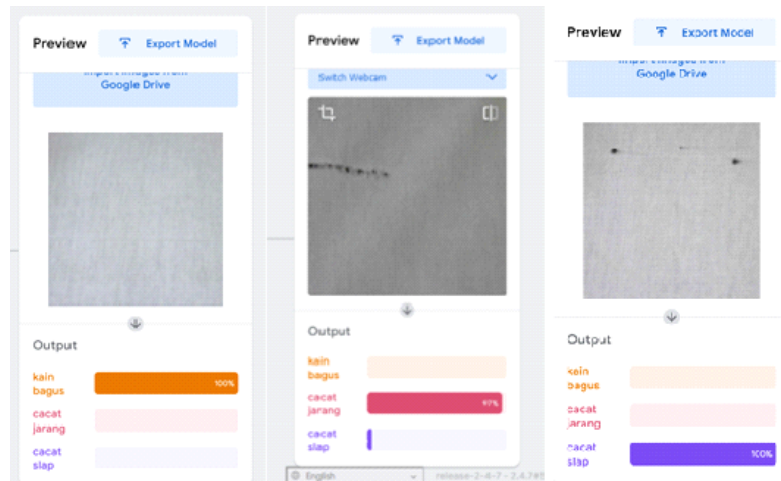
Gambar 7. Proses Training pada Teachable Machine

Pada Gambar 7 menampilkan proses training pada Teachable Machine, beberapa istilah penting dalam proses ini adalah

- 1) Epochs: Jumlah total data set yang digunakan dalam model training. Jumlah epoch menentukan berapa banyak iterasi yang diperlukan selama proses training.
- 2) Ukuran Batch: Jumlah data set yang diproses dalam iterasi training. Mempengaruhi kecepatan training dan penggunaan sumber daya komputasi.
- 3) Learning Rate: Parameter yang mengontrol seberapa banyak parameter model berubah selama training.

4.4 Evaluation

Tahap evaluasi digunakan untuk menguji nilai klasifikasi. Pada teachable machine, tahap evaluasi ditentukan di bagian "Preview". Tahap Preview ditunjukkan pada Gambar 8.

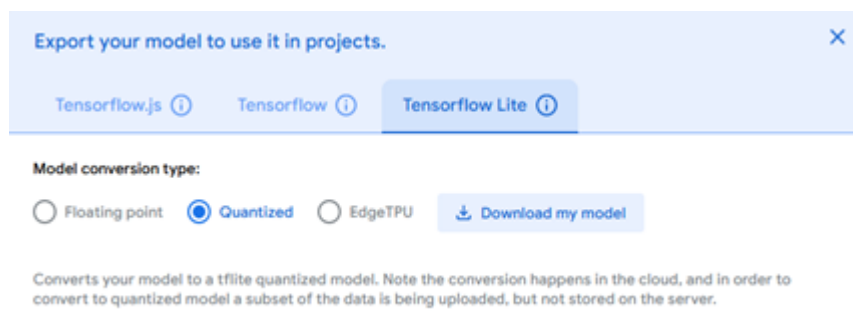


Gambar 8. Hasil Initial Classification

Berdasarkan Gambar 8, ketika kamera diberikan model, hasil Preview menunjukkan output dengan tingkat kepercayaan kecocokan data sebesar 97% hingga 100%. Dari data nilai hasil evaluasi model klasifikasi citra untuk setiap kategori yang dibuat, proses evaluasi menunjukkan hal yang positif, dimana nilai klasifikasi pada setiap kategori menunjukkan nilai yang cukup tinggi dan model telah mampu menunjukkan performa yang baik

4.5 Export Model

Tahap ekspor model dalam Teachable Machine adalah proses mengambil data training yang telah dilatih dan mengubahnya ke dalam format yang sesuai dengan mikroprosesor yang digunakan. Pada implementasi ini, format yang digunakan adalah *tensorflow lite* (quantised), format ini dipilih karena mikroprosesor yang digunakan adalah Raspberry PI-4. Model tensorflow lite yang dikuantisasi merupakan model machine learning yang dioptimalkan untuk digunakan pada perangkat berdaya rendah, dengan ukuran dan kompleksitas model yang lebih efisien, lebih cepat, dan cocok untuk sistem real-time. Gambar 9 menunjukkan proses ekspor model di Teachable Machine.

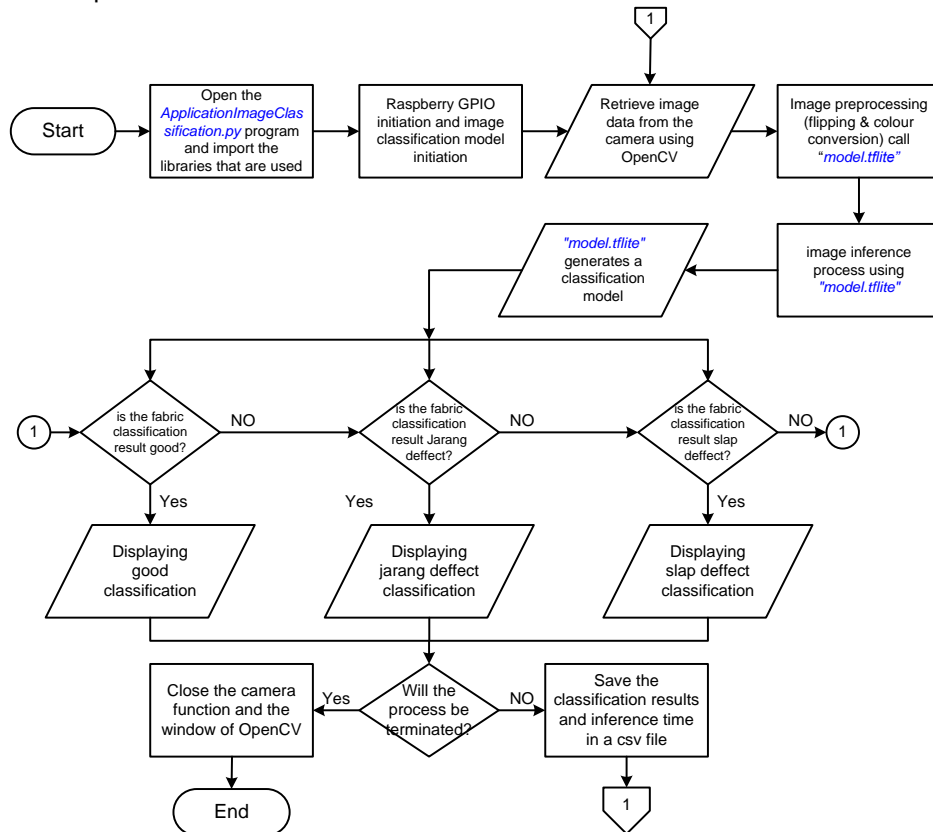


Gambar 9. Tahap Export model Pada Teachable Machine

Setelah model berhasil diekspor, seperti yang ditunjukkan pada Gambar 9, maka akan menghasilkan tensorflow dalam 3 format, yang dapat dipilih sesuai dengan mikroprosesor yang digunakan. Hasil dari ekspor tersebut berupa dua buah file, yaitu file dalam format `.tflite` (model.tflite) dan label dalam format `.text` (labels.txt). Label tersebut berupa informasi nama kategori atau kelas.

4.6 Deployment

Tahap deployment adalah implementasi tensorflow lite pada mikroprosesor Raspberry PI dengan menggunakan aplikasi python `ApplicationImageClassification.py`. Tahap implementasi ini dijelaskan oleh diagram alir pada Gambar 10.



Gambar 10. Diagram Alir Program ApplicationImageClassification.py

Berdasarkan Gambar 10, proses ini dimulai dengan inialisasi sistem. Ini termasuk mengkonfigurasi pin GPIO pada Raspberry Pi untuk mengontrol perangkat keras seperti relay dan LED. Selanjutnya, program mengambil gambar dari sumber kamera dan melakukan pra-pemrosesan yang diperlukan seperti membalik dan konversi warna. Gambar yang diambil dari kamera dioperasikan dengan model image classification yang telah diinisialisasi sebelumnya. Hasil inferensi dari model tersebut digunakan untuk melakukan klasifikasi pada gambar. Jika hasil klasifikasi memenuhi kondisi tertentu yang sudah ditentukan sebelumnya, program akan mengontrol pin GPIO berdasarkan hasil klasifikasi, termasuk mengatur relay dan LED menurut kondisi yang terdeteksi. Proses ini terus berlanjut secara terus-menerus, kembali ke langkah menangkap gambar dari kamera setelah langkah klasifikasi dan kontrol perangkat keras selesai. Jika pengguna menekan tombol ESC, program akan berhenti dan menutup

4.7 Pengujian Hasil

Dalam proses pengujian hasil, dilakukan menggunakan library Open CV untuk nilai akurasi dan hasil performa sistem yang diperoleh melalui library berikut:

1. `value_fps.csv`. File ini digunakan untuk mencari data uji performa. File ini digunakan untuk menyimpan data FPS (Frame Per Second), yang mencerminkan seberapa cepat aplikasi mampu melakukan klasifikasi pada setiap frame gambar. Informasi yang disimpan meliputi cap waktu, waktu pengujian dilakukan dan nilai FPS pada saat itu.

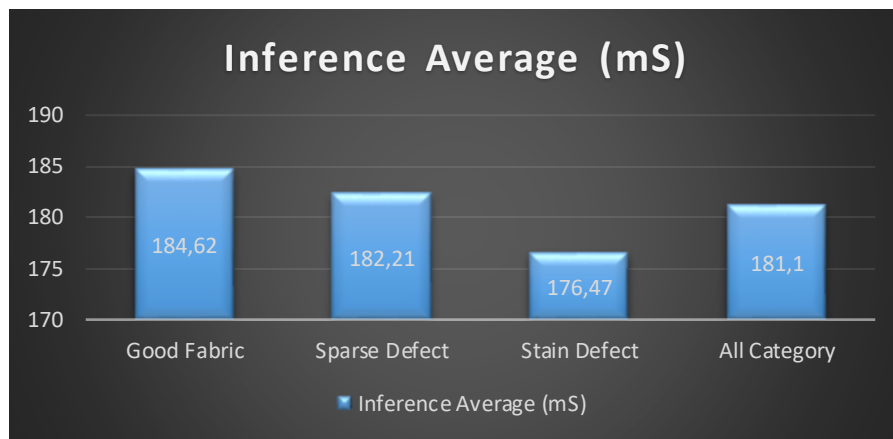
2. Hasil_klasifikasi.csv': File ini digunakan untuk mencari data uji akurasi klasifikasi model. File ini menyimpan hasil klasifikasi yang didapatkan oleh model pada setiap iterasi atau frame selama aplikasi dijalankan. Setiap baris pada file ini berisi informasi seperti timestamp (waktu klasifikasi dilakukan), nama kategori yang diprediksi oleh model, dan skor hasil klasifikasi dengan menyimpan data ini dalam format CSV.

4.7.1 Pengujian performa

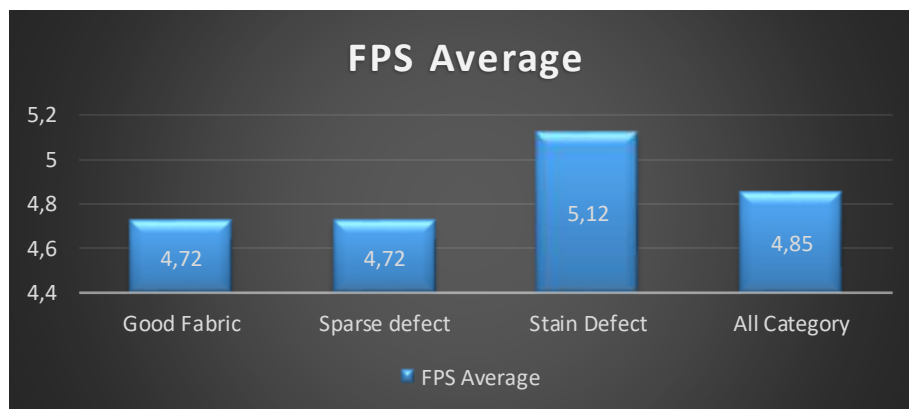
Uji performa adalah pengukuran kecepatan operasi dari 'Penerapan Image classification', yang diukur dalam satuan FPS (frame per detik), dan pengujian inferensi pada setiap kategori. Data yang dikumpulkan adalah hasil data dari proses sebelumnya dalam bentuk file 'value_fps.csv'. Tabel 2 dan Gambar 11-15 menunjukkan nilai data waktu inferensi dan parameter fps untuk setiap kategori.

Tabel 2 Hasil performa sistem Image Classification

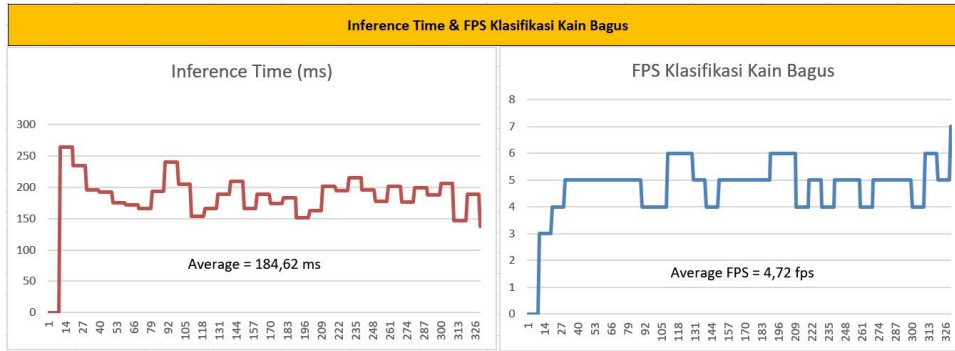
Kategori	Rata-rata Inferensi (mS)	Rata-rata FPS
Kain bagus	184,62	4,72
Kain Cacat Jarang	182,21	4,72
Kain Cacat slap	176,47	5,12
Semua kategori	181,1	4,85



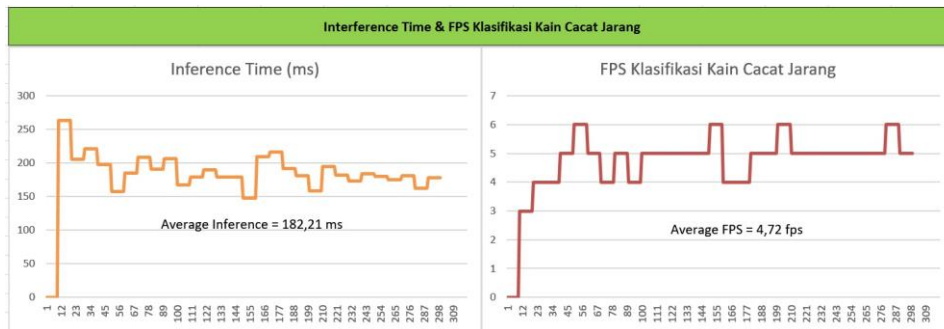
Gambar 11 Hasil Kinerja Inferensi Aplikasi Klasifikasi Citra



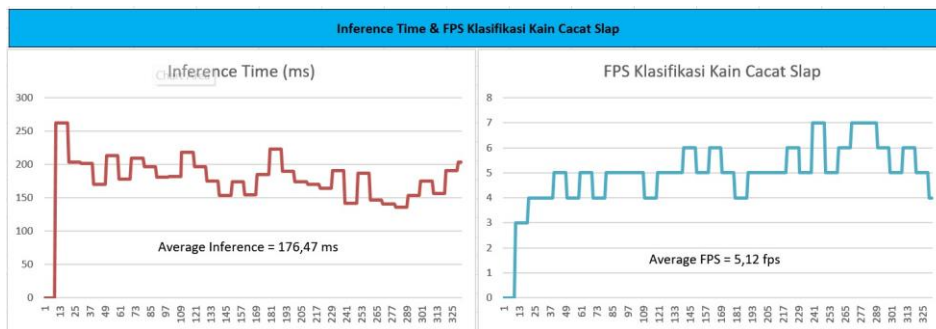
Gambar 12 Hasil Kinerja FPS Aplikasi Image classification



Gambar 13 Grafik Performa Aplikasi Image classification Kain yang Bagus



Gambar 14 Grafik Performa Aplikasi Klasifikasi Citra Cacat Jarang



Gambar 15 Grafik Kinerja Aplikasi Klasifikasi Citra Cacat Noda

Dari hasil yang ditunjukkan pada Gambar 11-15, model klasifikasi dapat menghasilkan klasifikasi dengan waktu inferensi rata-rata 181,1 ms di semua kategori dan FPS rata-rata 4,85 FPS di semua kategori. Meskipun terdapat sedikit perbedaan waktu inferensi dan FPS antara klasifikasi kain cacat tampak dengan yang lainnya, secara umum model dapat mengklasifikasikan gambar dengan respon yang baik. Kecepatan FPS yang konsisten di seluruh klasifikasi menunjukkan konsistensi dalam kinerja aplikasi.

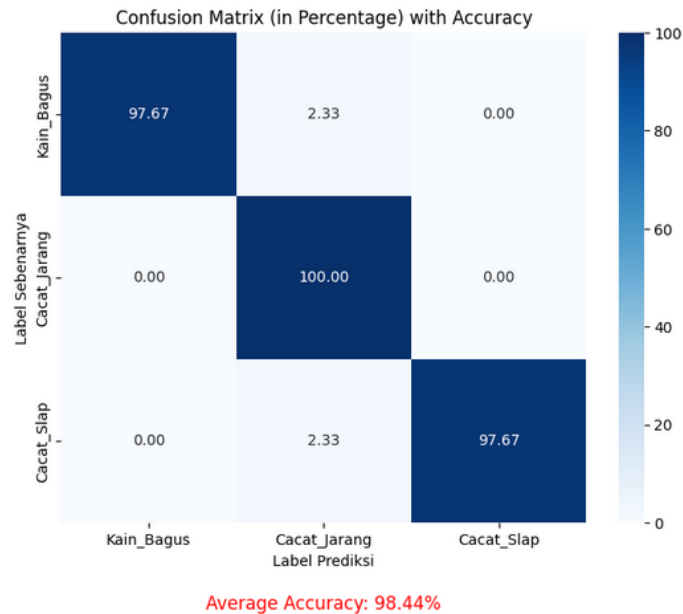
4.7.2 Pengujian akurasi Klasifikasi model

Pengujian akurasi klasifikasi model mengevaluasi sejauh mana model image classification dapat mengidentifikasi objek atau atribut dalam gambar dengan benar. Hasilnya disajikan dalam bentuk tabel atau grafik confusion matrix untuk mengetahui kemampuan akurasi model. Data yang diambil merupakan hasil dari pengambilan data pada proses sebelumnya, yaitu berupa file 'Hasil_Klasifikasi.csv'. Tabel 3 menunjukkan data yang telah diambil dari file CSV tersebut.

Tabele 3 Hasil Klasifikasi Data

PREDIKSI	ACTUAL		
	Kain Bagus	Cacat Jarang	Cacat Slap
Kain Bagus	293	0	0
Cacat Jarang	7	300	7
Cacat Slap	0	0	293

Berdasarkan Tabel 3 menjelaskan bahwa pengujian kain bagus selama 184.62 mS berhasil membaca 300 hasil inferensi dengan 293 dinyatakan sebagai kain bagus dan 7 dinyatakan sebagai cacat jarang. Pada pengujian cacat jarang sebesar 182.21 mS terbaca 300 hasil inferensi dengan 300 dinyatakan sebagai cacat jarang. Pada pengujian cacat slap sebesar 176.47 mS, 300 hasil inferensi terbaca dengan 293 dinyatakan sebagai cacat slap dan 7 dinyatakan sebagai cacat jarang. Gambar 15 adalah visualisasi dari confusion matrix dalam bentuk persentase, berdasarkan data yang terdapat pada Tabel 3.



Gambar 16 Tampilan Confusion Matrix dalam Persentase

Berdasarkan Gambar 16, dapat disimpulkan bahwa rata-rata akurasi dari seluruh hasil uji coba klasifikasi model memperoleh nilai akurasi sebesar 98,44%.

5 Kesimpulan

Berdasarkan hasil dan pembahasan di atas, penelitian ini dapat menarik kesimpulan sebagai berikut

1. Pemrograman Python telah mampu membangun sebuah aplikasi Klasifikasi Citra yang memungkinkan hasil tangkapan kamera dapat dilihat oleh pengguna secara real time dan terus menerus.
2. Performa Aplikasi Klasifikasi Citra untuk mengklasifikasikan kecacatan kain menghasilkan rata-rata waktu inferensi pada semua kategori sebesar 181.1 ms dan rata-rata FPS pada semua kategori sebesar 4.85 FPS dengan tingkat akurasi sebesar 98.44%.
3. Pemanfaatan TensorFlow Lite pada Raspberry Pi terbukti sangat cocok dan efisien dalam menjalankan model. TensorFlow Lite memungkinkan kinerja yang lebih responsif dan efisien dalam sumber daya yang terbatas pada perangkat Raspberry Pi.
4. Penggunaan Teachable Machine dalam mengembangkan model Machine Learning tidak menunjukkan tanda-tanda overfitting. Hal ini mengindikasikan bahwa pengalaman dalam membangun model menggunakan Teachable Machine mampu menghasilkan model yang menggeneralisasi dengan baik dengan data baru tanpa menyerah pada kondisi overfitting, sehingga model dapat diterapkan secara efektif di berbagai skenario pengenalan objek atau klasifikasi.

Referensi

- Abou Baker, N., Zengeler, N., & Handmann, U. (2022). A Transfer Learning Evaluation of Deep Neural Networks for Image Classification. *Machine Learning and Knowledge Extraction*, 4(1), 22–41. <https://doi.org/10.3390/make4010002>
- Abu, M. A., Indra, N. H., Rahman, A. H. A., Sapiee, N. A., & Ahmad, I. (2019). A study on image classification based on deep learning and tensorflow. *International Journal of Engineering*

- Research and Technology*, 12(4), 563–569.
- Alaskar, H., & Saba, T. (2021). *Machine Learning and Deep Learning: A Comparative Review*. June, 143–150. https://doi.org/10.1007/978-981-33-6307-6_15
- Bandyopadhyay, H. (2022). *What Is Computer Vision? [Basic Tasks & Techniques]*. V7.
- Chazar, C., & Rafsanjani, M. H. (2022). *LPPM STMIK ROSMA / Prosiding Seminar Nasional : Inovasi & Adopsi Teknologi Penerapan Teachable Machine Pada Klasifikasi Machine Learning Untuk Identifikasi Bibit Tanaman*.
- Dewanti, Firma, D., & Pujotomo, D. (2017). Analisis Penyebab cacat Produk Kain dengan Menggunakan Metode FMEA. *Jurnal Teknik Industri*, 6(4), 1–7.
- Dompeipen, T. A., & Sompie, S. R. U. . (2020). Penerapan computer vision untuk pendeteksian dan penghitung jumlah manusia. *Jurnal Teknik Informatika*, 15(4), 1–12.
- Han, S.-H., Kim, K. W., Kim, S., & Youn, Y. C. (2018). Artificial Neural Network: Understanding the Basic Concepts without Mathematics. *Dementia and Neurocognitive Disorders*, 17(3), 83. <https://doi.org/10.12779/dnd.2018.17.3.83>
- Kamavisdar, P., Saluja, S., & Agrawal, S. (2013).). *A survey on image classification approaches and techniques*. *Nternational Journal of Advanced Research in Computer and Communication Engineering*. 2(1), 1005–1009.
- Kementerian Perindustrian. (2019). *Kemenperin: Industri Tekstil dan Pakaian Tumbuh Paling Tinggi*. Kementerian Perindustrian.
- Kementerian Perindustrian. (2021). Mendorong Kinerja Industri Tekstil dan Produk Tekstil di Tengah Pandemi. *Buku Analisis Pembangunan Industri*, 1–37.
- Kristanto, A. Y., Rumita, R., & Sriyanto. (2016). Analisis Penyebab Cacat Kain dengan Menggunakan Metode Failure Mode and Effect Analysis (FMEA) dan Fault Tree Analysis (FTA). *Industrial Engineering Online Journal*, 5(1), 1–8.
- Malahina, E. A. U., Hadjon, R. P., & Bisilisin, F. Y. (2022). Teachable Machine: Real-Time Attendance of Students Based on Open Source System. *The IJICS (International Journal of Informatics and Computer Science)*, 6(3), 140. <https://doi.org/10.30865/ijics.v6i3.4928>
- Maulana, I. (2021). Implementasi Raspberry Pi 4 Sebagai Server E-Learning. *Jurnal Media Aplikom*, 13(2), 53–67.
- Pintelas, E., Liaskos, M., Livieris, I. E., Kotsiantis, S., & Pintelas, P. (2020). Explainable machine learning framework for image classification problems: Case study on glioma cancer prediction. *Journal of Imaging*, 6(6). <https://doi.org/10.3390/JIMAGING6060037>
- Prasad, P. Y., Prasad, D., Malleswari, N., Shetty, M. N., & Gupta, N. (2022). Implementation of Machine Learning Based Google Teachable Machine in Early Childhood Education. *International Journal of Early Childhood Special Education*, 14(3), 4132–4138.
- Pratama, Y., Lestari, U., & Hamzah, A. (2022). *Pemanfaatan Aplikasi Teachable Machine Untuk Pengenalan Binatang Menggunakan Konsep Convolutional Neural Network (Cnn)*. 10(1), 10–20.
- Purno, A., & Wibowo, W. (2016). Implementasi Teknik Computer Vision Dengan Metode Colored Markers Trajectory Secara Real Time. *Jurnal Teknik Informatika*, 8(1), 38–42.
- Setiawan, F. B., Kusuma, H. W., Riyadi, S., & Pratomo, Leonardo H. (2022). Penerapan PI Cam Menggunakan Program Berbasis Raspberry PI 4. *CYCLOTRON: Jurnal Teknik Elektro*, 5(2), 51–56.
- Wijaya, I. D., Nurhasan, U., & Barata, M. A. (2017). Implementasi Raspberry Pi Untuk Rancang Bangun Sistem Keamanan Pintu Ruang Server Dengan Pengenalan Wajah Menggunakan Metode Triangle Face. *Jurnal Informatika Polinema*, 4(1), 9. <https://doi.org/10.33795/jip.v4i1.138>