

BAB I PENDAHULUAN

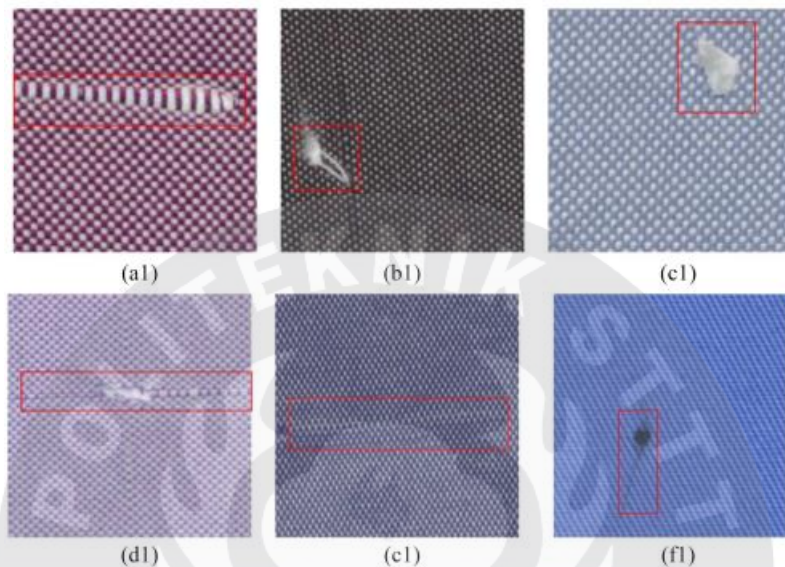
1.1 Latar Belakang

Industri tekstil merupakan salah satu sektor manufaktur terbesar di dunia, yang memainkan peran penting dalam perekonomian banyak negara. Namun, kualitas produk tekstil sering kali terganggu oleh cacat kain yang dapat terjadi selama proses produksi. Cacat kain seperti pakan melintir, lusi putus, pakan kosong dan lainnya dapat mengurangi nilai jual produk dan meningkatkan biaya produksi akibat kebutuhan untuk inspeksi ulang dan perbaikan. Oleh karena itu diperlukan proses inspeksi untuk melakukan *grading* kain.

Teknik konvensional untuk inspeksi kain umumnya bersifat manual dan sangat bergantung pada keahlian operator manusia. Metode ini tidak hanya memakan waktu dan tenaga, tetapi juga rentan terhadap kesalahan manusia, yang dapat menyebabkan cacat terlewatkan atau salah klasifikasi (Jing et al., 2019). Dalam upaya untuk mengatasi keterbatasan ini, teknologi otomatisasi berbasis komputer telah mulai diterapkan. Salah satu teknologi yang menjanjikan adalah *Convolutional Neural Networks* (CNN), sebuah arsitektur *deep learning* yang telah terbukti sangat efektif dalam berbagai tugas pengenalan pola dan klasifikasi gambar. CNN telah menunjukkan hasil yang mengesankan dalam bidang-bidang seperti pengenalan wajah, deteksi objek, dan pengklasifikasian gambar. Kemampuan CNN untuk secara otomatis belajar fitur-fitur penting dari data gambar menjadikannya sebuah pilihan (Akram et al., 2023). Namun, penerapan CNN untuk klasifikasi cacat kain masih relatif baru dan memerlukan penelitian lebih lanjut untuk mengevaluasi efektivitasnya dalam konteks ini.

Penelitian ini bertujuan untuk mengevaluasi apakah CNN dapat digunakan secara efektif untuk mendeteksi dan mengklasifikasikan berbagai jenis cacat kain. Dengan menggunakan teknik transfer learning, yaitu dengan menggunakan model pra-latih, model CNN yang telah dikembangkan pada dataset gambar umum dapat disesuaikan untuk tugas spesifik ini, sehingga memanfaatkan kemampuan model pra-latih tersebut dalam mengenali fitur dasar gambar (Jia et al., 2022). Penelitian ini akan menguji kemampuan model dalam mengidentifikasi dan mengklasifikasikan cacat kain.

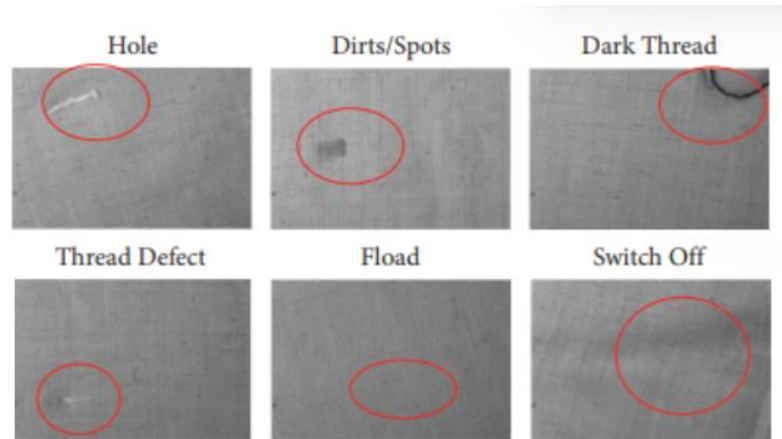
Penelitian yang berkaitan dengan penggunaan CNN dalam mengidentifikasi, klasifikasi dan deteksi cacat pada kain sudah dilakukan dalam penelitian lain. Penggunaan *Transfer Learning* dari berbagai arsitektur diuji dan di bandingkan, dan augmentasi pada dataset sebelum dilakukan pelatihan . Arsitektur Resnet50 menghasilkan akurasi tinggi mencapai hampir 95% pada 6 jenis cacat yang dapat dilihat pada gambar 1.1 (Jia et al., 2022).



Sumber : *Fabric defect detection based on transfer learning and improved Faster R-CNN*, (Jia et al., 2022b)

Gambar 1. 1 Jenis cacat pada penelitian sebelumnya : (a1) *ribbon yarn*, (b1), *broken yarn*, (c1) *cotton ball*, (d1) *hole*, (e1) *yarn shedding*, (f1) *stain*.

Pada penelitian serupa lainnya menggunakan arsitektur googlenet untuk mengekstrak fitur-fitur yang kaya dan mengklasifikasikan sampel kain ke dalam berbagai kategori cacat diantara-Nya *hole*, *spot*, *dark thread*, *thread defect*, *float*, dan *switch off* yang dapat dilihat pada gambar 1.2. Sistem ini dikembangkan dan dievaluasi pada dataset TILDA, dan mencapai akurasi klasifikasi keseluruhan sebesar 94,46%, mengungguli teknik-teknik lainnya seperti SVM dan BPNN yang dicantumkan pada jurnal tersebut.



Sumber : *Classification of Woven Fabric Faulty Images Using Convolution Neural Network* (Ashraf et al., 2022)

Gambar 1. 2 Jenis jenis cacat pada penelitian

Pada penelitian lainnya menerapkan algoritma 2 tahap, di mana tahap pertama model hanya dikembangkan untuk mengenali cacat dan tidak cacat, kemudian tahap kedua model dikembangkan untuk mengenali berbagai jenis cacat yang ada. Dengan menggunakan model arsitektur LeNet-5. Jenis cacat pada jurnal ini jenis cacat dibagi dengan 3 kategori , cacat tanpa arah, cacat ke arah lusi dan cacat ke arah pakan. Hasil penelitian ini dapat dilihat pada gambar 1.3.

Metrics	Without pre-processing	With grayscale stretching	With proposed equalization
Precision			
Normal	92.5%	92.7%	93.7%
No direction	93.3%	93.5%	93.8%
Warp direction	88.6%	91.1%	91.7%
Weft direction	87.6%	89.2%	90.4%

Sumber : *Fabric defect detection based on a deep Convolutional neural Network using a two-stage strategy* (Jun et al., 2021).

Gambar 1. 3 Akurasi model CNN

Tantangan dalam menggunakan CNN untuk klasifikasi cacat kain terdapat pada pengenalan cacat yang berukuran kecil, konstruksi kain yang berbeda beda hingga kecepatan dan efisiensi algoritma model yang berhubungan dengan bagaimana meminimalkan waktu deteksi sambil mempertahankan akurasi tinggi (Jun et al., 2021).

Penelitian ini menggunakan arsitektur CNN ternama, yaitu VGG16, dengan 5 kategori cacat kain serta satu kategori kain tanpa cacat. Salah satu fokus

penelitian ini adalah untuk menentukan jumlah *epoch* optimal bagi tiap variasi *pooling layer* tambahan: *Maxpooling*, *Averagepooling*, dan tanpa *pooling layer* tambahan. *Epoch* mengacu pada jumlah iterasi lengkap model dalam memproses seluruh data pelatihan, di mana setiap *epoch* dapat mempengaruhi akurasi dan performa model secara keseluruhan. Oleh karena itu, pemilihan jumlah *epoch* yang tepat sangat penting untuk mencapai keseimbangan antara pelatihan yang cukup dan menghindari *overfitting*. Untuk setiap variasi, dilakukan percobaan dengan 10, 25, dan 50 *epoch*. Model yang memiliki akurasi terbaik akan diimplementasikan ke dalam TensorFlow Lite untuk Android guna mengamati performa model secara *real-time*. Selain itu, model ini juga akan diimplementasikan pada platform Gradio untuk melihat performa model secara *delayed*, tanpa kompresi seperti pada TensorFlow Lite. Gradio adalah antarmuka web berbasis Python yang memudahkan pengguna untuk menguji model *machine learning* secara interaktif melalui antarmuka grafis tanpa perlu membuat kode tambahan yang kompleks. Dengan Gradio, pengguna dapat mengunggah gambar, melihat hasil prediksi, dan berinteraksi langsung dengan model dalam lingkungan web.

Hasil dari penelitian ini diharapkan dapat memberikan wawasan yang berharga mengenai potensi penggunaan CNN dalam industri tekstil, serta membuka jalan bagi pengembangan sistem inspeksi kain otomatis yang lebih efisien dan akurat. Maka dari itu akan dilakukan penelitian yang disajikan dalam bentuk skripsi yang berjudul :

“STUDI PENGGUNAAN CONVOLUTIONAL NEURAL NETWORKS DALAM DETEKSI DAN KLASIFIKASI CACAT KAIN ”

1.2 Identifikasi Masalah

Dalam konteks penelitian ini, masalah utama yang diidentifikasi adalah:

- 1) Berapa jumlah *epoch* yang optimal untuk tiga variasi *tuning* model, dan kombinasi mana yang terbaik?
- 2) Bagaimana performa model saat diimplementasikan ke tensorflow lite?
- 3) Apakah ada perbedaan akurasi saat model di implementasikan di tensorflow lite dibandingkan dengan gradio?

1.3 Batasan Masalah

Karena keterbatasan waktu dan sumber daya yang ada, juga untuk memastikan fokus yang jelas dan kelayakan dalam pelaksanaannya. Maka batasan-batasan masalah yang akan dianalisis antara lain :

1. Penelitian ini hanya mencakup 5 jenis cacat kain yaitu *snarl*, lusi putus, pakan hilang, pakan bulu, pakan dobel dan kain dengan kategori bagus. Pembatasan ini dilakukan karena keterbatasan waktu dan kemampuan peneliti dalam pengumpulan dataset.
2. Dataset yang digunakan dalam penelitian ini terdiri dari gambar kain yang dikumpulkan secara langsung, dengan jumlah gambar sekitar 300 per jenis cacat. Dataset yang dikumpulkan adalah foto kain greige yang diproduksi dan diinspeksi di PT.Sipatex Puteri Lestari.
3. Foto kain greige pada penelitian ini merupakan foto dari beberapa jenis konstruksi kain di PT.Sipatex Puteri Lestari dengan jenis anyaman: *ripstop*, polos, marsoto.
4. Penelitian ini menggunakan *Convolutional Neural Networks* (CNN) sebagai metode utama untuk deteksi dan klasifikasi cacat kain menggunakan *framework* TensorFlow.
5. Model yang digunakan akan memanfaatkan teknik transfer learning dengan model pra-latih VGG16 untuk meningkatkan kinerja dengan dataset yang terbatas.
6. Evaluasi model akan dilakukan berdasarkan metrik-metrik standar yaitu akurasi dan validasi.
7. Pengujian model akan dilakukan pada dataset yang telah dipisahkan secara khusus untuk keperluan pengujian dan validasi.
8. Implementasi hasil penelitian dibatasi pada hasil uji dan validasi model CNN yang telah dibuat sebagai tolak ukur penelitian juga implementasi untuk Tensorflow Lite android dan Gradio. Penggunaan dalam skala industri atau pengujian pada lingkungan produksi nyata tidak termasuk dalam lingkup penelitian ini.

Dengan batasan-batasan ini, penelitian diharapkan dapat dilakukan secara fokus dan mendalam, menghasilkan temuan yang spesifik dan relevan mengenai penggunaan CNN untuk deteksi cacat kain.

1.4 Maksud dan Tujuan

Penelitian ini dimaksudkan untuk mengeksplorasi dan mengevaluasi penggunaan *Convolutional Neural Networks* (CNN) dalam mendeteksi dan mengklasifikasikan cacat kain. Dengan memahami potensi dan keterbatasan teknik ini, penelitian ini bertujuan untuk memberikan dasar ilmiah bagi pengembangan sistem inspeksi kain otomatis.

1.5 Kerangka Pemikiran

Deteksi cacat kain adalah proses identifikasi dan klasifikasi ketidaksempurnaan pada kain yang dapat mempengaruhi kualitas produk. Cacat kain dapat bervariasi dalam bentuk dan ukuran, termasuk lubang, noda, benang putus, dan lain-lain (Jing et al., 2019). Salah satu metode yang diusung sebagai dasar dalam membuat sistem deteksi dan klasifikasi cacat kain adalah dengan menggunakan *Convolutional Neural Networks*.

Convolutional Neural Networks (CNN) adalah jenis jaringan saraf tiruan yang dirancang khusus untuk pemrosesan data *grid*, seperti gambar. CNN terdiri dari beberapa lapisan yang berbeda, termasuk lapisan konvolusi, *pooling*, dan *fully connected*. Struktur CNN memungkinkan model ini untuk secara otomatis belajar fitur penting dari data gambar, yang membuatnya sangat efektif untuk tugas-tugas pengenalan pola dan klasifikasi (Ian Goodfellow, Yoshua Bengio, 2017)

Arsitektur pada CNN terdiri dari :

- Lapisan Konvolusi (*Convolutional Layer*):
Fungsi utama lapisan ini adalah untuk mendeteksi fitur-fitur lokal dalam gambar melalui operasi konvolusi. Lapisan konvolusi menggunakan filter (kernel) yang bergerak melintasi gambar, menghasilkan peta fitur yang menunjukkan keberadaan fitur tertentu di berbagai lokasi. Setiap filter bertanggung jawab untuk mendeteksi fitur spesifik seperti tepi, tekstur, atau pola tertentu dalam gambar (O'Shea & Nash, 2015).
- Lapisan *Pooling* (*Pooling layer*):
Lapisan *pooling* bertujuan untuk mengurangi dimensi peta fitur, yang membantu mengurangi jumlah parameter dan komputasi dalam jaringan. *Pooling* juga membantu dalam membuat deteksi fitur menjadi lebih invariant terhadap pergeseran dan deformasi. Jenis *pooling* yang umum adalah *max pooling*, di mana nilai maksimum dalam setiap *patch* kecil dari peta fitur dipilih sebagai representasi (O'Shea & Nash, 2015).

Selain *max pooling*, lapisan *pooling* yang umum digunakan adalah *Average pooling* di mana nilai rata-rata dari peta fitur dipilih sebagai representasi.

Karena tidak adanya aturan baku tentang *pooling* yang harus digunakan, maka pada penelitian ini diujikan 2 jenis *pooling* sebagai *layer* tambahan, dan tanpa *layer* tambahan.

- Lapisan *Fully Connected* (*Fully Connected Layer*):

Lapisan ini menghubungkan semua neuron dari lapisan sebelumnya ke semua neuron di lapisan berikutnya. Lapisan *Fully connected* berfungsi untuk menggabungkan fitur-fitur yang telah dipelajari menjadi representasi yang lebih abstrak dan membuat prediksi akhir (O'Shea & Nash, 2015).

Dalam penelitian ini, berbagai variasi *pooling layer*, seperti *Maxpooling* dan *Averagepooling*, diuji untuk menentukan pengaruhnya terhadap performa model. *Pooling layer* berfungsi untuk mengurangi dimensi data dan meningkatkan efisiensi pemrosesan, namun tidak ada aturan baku mengenai jenis *pooling* yang harus digunakan. Oleh karena itu, penelitian ini mengevaluasi beberapa jenis *pooling* serta konfigurasi tanpa *pooling layer* tambahan untuk menentukan yang terbaik untuk dataset cacat kain.

Untuk menghemat waktu komputasi dan memperkuat model dengan dataset yang terbatas, diterapkan teknik transfer learning. Transfer learning adalah teknik dalam *machine learning* di mana model yang telah dikembangkan pada satu tugas digunakan kembali untuk tugas lain yang terkait. Dalam konteks CNN, model pra-latih yang telah dikembangkan pada dataset gambar besar seperti ImageNet dapat digunakan sebagai dasar untuk pelatihan pada dataset yang lebih spesifik, seperti gambar cacat kain (Jia et al., 2022). Keuntungan Transfer Learning:

- Penghematan Waktu dan Sumber Daya: Menggunakan model pra-latih mengurangi kebutuhan untuk melatih model dari awal, yang memerlukan waktu dan sumber daya komputasi yang besar (Pan & Yang, 2010).
- Kinerja yang Lebih Baik: Model pra-latih telah belajar fitur-fitur umum dari dataset besar, yang dapat membantu dalam mendeteksi pola pada dataset yang lebih kecil dan spesifik (Yosinski et al., 2014).
- Mengatasi Keterbatasan Data: Transfer learning memungkinkan

penggunaan dataset yang lebih kecil dengan tetap mendapatkan kinerja yang baik, karena model sudah memiliki pengetahuan awal dari dataset besar (Weiss et al., 2016)

Arsitektur yang digunakan pada penelitian ini adalah VGG16. VGG Net mencapai tingkat kesalahan sebesar 7,32% dalam ILSVRC 2014 dan menjadi model *runner-up* dalam ILSVRC 2014 .

Namun, efektivitas model tidak hanya dipengaruhi oleh arsitektur dan teknik transfer learning, tetapi juga oleh parameter pelatihan, salah satunya adalah jumlah *epoch*. *Epoch* menandakan satu siklus algoritma *machine learning* 'belajar' dari seluruh set data *training*. Satu *epoch* berarti sebuah algoritma *machine learning* telah 'belajar' dari data *training* secara keseluruhan. Dalam jaringan saraf tiruan, proses pembelajaran yang berulang-ulang bertujuan untuk mencapai konvergensi nilai bobot (Wibawa, 2017). Namun jumlah *epoch* dan kualitas model tidak selalu berbanding (Resa Arif Yudianto & Al Fatta, 2020). Jumlah *epoch* yang terlalu besar, dapat menyebabkan *overfitting* pada model. *Overfitting* adalah kondisi di mana hampir semua data yang telah melalui proses *training* mencapai persentase yang baik, tetapi terjadi ketidaksesuaian pada proses prediksi (Santoso, 2018).

Oleh karena itu, penelitian ini menguji variasi jumlah *epoch*—10, 25, dan 50—untuk menemukan keseimbangan optimal yang menghindari *overfitting* sambil memastikan pembelajaran yang memadai. Pemilihan variasi jumlah *epoch* pada penelitian ini didasarkan pada kemampuan komputer yang digunakan dalam melakukan komputasi proses pelatihan model, juga merujuk dari penelitian terdahulu. Pada penelitian berjudul *Fabric Defect Detection Based on Lightweight Neural Network* (Grabisch et al., 2019), nilai *epoch* yang digunakan adalah 50 *epoch*.

Dalam konteks implementasi, model diuji dengan TensorFlow Lite (TFLite) dan Gradio. TensorFlow Lite dirancang untuk optimasi model pada perangkat seluler dan IoT. TensorFlow Lite sendiri mempunyai keunggulan yaitu mempunyai sistem optimasi yang membuat proses klasifikasi menjadi lebih ringan pada perangkat Mobile (Wijaya, 2023). Tensorflow Lite bekerja dengan mengubah model Tensorflow yang sudah jadi menjadi model TFLite, membuat proses klasifikasi menjadi lebih ringan dan dengan ukuran file yang lebih kecil. Komputasi pada TFLite telah direduksi agar menjadi lebih cepat dan kompatibel dengan *microdevice* (Wijaya, 2023). Sebaliknya, Gradio menyediakan antarmuka

interaktif yang memungkinkan evaluasi model tanpa optimasi tambahan, sehingga dapat memberikan wawasan mengenai performa model dalam kondisi yang lebih mendekati implementasi awal. Gradio adalah pustaka Python sumber terbuka yang mempermudah proses pembuatan antarmuka pengguna untuk model *machine learning*. Gradio menyediakan API tingkat tinggi untuk membangun antarmuka interaktif untuk model *machine learning* (E et al., 2024). Secara keseluruhan, kerangka pemikiran ini menunjukkan bahwa terdapat kemungkinan perbedaan akurasi antara implementasi TFLite dan Gradio karena proses optimasi yang diterapkan oleh TFLite. Jumlah *epoch* yang optimal tidak selalu dapat diperkirakan dengan pasti, tetapi penelitian ini bertujuan untuk mengidentifikasi titik keseimbangan yang menghindari *overfitting* sambil memastikan performa model yang baik.

1.6 Metodologi Penelitian

Metodologi yang dilakukan pada penelitian ini adalah sebagai berikut:

1) Studi Literatur

Penelusuran pustaka yang berkaitan dengan penelitian yang dilakukan. Berupa membaca dan mengumpulkan jurnal atau skripsi artikel dan buku yang berhubungan dengan penelitian guna memperoleh ide, pengetahuan yang dibutuhkan, mencari ide yang cocok dan sebagai referensi.

2) Pengumpulan Data

- Gambar kain dengan cacat dipotret menggunakan kamera *smartphone* dengan pencahayaan yang seragam untuk mengurangi variasi yang disebabkan oleh kondisi eksternal.
- Setiap jenis cacat akan memiliki sekitar 300 gambar, sehingga total dataset sebelum dilakukan augmentasi mencakup 1500 gambar. Gambar akan disimpan dalam format JPEG dengan resolusi yang cukup untuk mendeteksi cacat .

3) Pra-proses Data

- Transformasi dan Normalisasi:

Gambar akan diubah ukurannya menjadi seragam, ukuran disesuaikan dengan kebutuhan *input* untuk arsitektur yang digunakan yaitu 224x224. Normalisasi akan dilakukan dengan mengubah nilai piksel menjadi rentang [0, 1] dengan membagi setiap nilai piksel dengan 255.

- Augmentasi Data:

Augmentasi data akan diterapkan untuk memperbesar variasi gambar. Teknik augmentasi meliputi rotasi, *flipping*, pemotongan, perubahan skala dan pencahayaan. Augmentasi data menghasilkan 2000 gambar yang berbeda dengan gambar asli, sehingga jumlah gambar menjadi 2300 gambar per kategori.

4) Pengembangan Model CNN

- Arsitektur CNN:

Model CNN dikembangkan dengan beberapa lapisan konvolusi, *pooling*, dan *Fully connected*. Arsitektur yang digunakan adalah arsitektur milik VGG16 kemudian dibuat 3 variasi yaitu dengan penambahan *layer max pooling*, *Average pooling* dan tanpa tambahan *layer*.

- Transfer Learning:

Model pra-latih VGG16 digunakan sebagai dasar untuk transfer learning. Lapisan atas model pra-latih dibekukan, dan lapisan-lapisan tambahan ditambahkan untuk tugas spesifik deteksi cacat kain.

5) Pelatihan dan Validasi Model

- Pembagian dataset:

Dataset akan dibagi menjadi data pelatihan (70%), data validasi (15%), dan data pengujian (15%). Data pelatihan akan digunakan untuk melatih model, data validasi untuk *tuning hyperparameter*, dan data pengujian untuk evaluasi akhir.

- Proses pelatihan:

Model dikembangkan menggunakan algoritma optimasi *A Method For Stochastic Optimization* (ADAM). *Loss function* yang digunakan adalah *categorical cross-entropy*, mengingat tugas klasifikasi Multi kelas. Pelatihan akan dilakukan dalam tiga variasi *epoch* untuk setiap variasi *pooling* yaitu *epoch* 10, 25 dan 50, dengan *batch size* 32. Pemilihan *batch size* digunakan berdasarkan kemampuan kinerja komputer yang digunakan.

6) Evaluasi Kinerja Model

- Akurasi (*Accuracy*):

Persentase prediksi yang benar dari total prediksi yang dibuat oleh model, akurasi akan dibandingkan antara arsitektur VGG16 dengan tambahan *max pooling*, *average pooling* dan tanpa *pooling layer* tambahan untuk ke tiga variasi *epoch*.

- Validasi

Validasi adalah proses untuk mengevaluasi kinerja model pada data yang tidak digunakan selama pelatihan, guna mengukur seberapa baik model tersebut akan bekerja pada data baru yang belum pernah dilihat sebelumnya. Validasi akan dibandingkan antara arsitektur VGG16 dengan tambahan *max pooling*, *average pooling* dan tanpa *pooling layer* tambahan untuk ke tiga variasi *epoch*.

- Model dengan akurasi terbaik kemudian diimplementasikan ke android dengan tensorflow lite, untuk dilihat hasil prediksi model secara *realtime*. Di gradio, model akan diimplementasikan di web yang di jalankan pada *localhost*, pada antarmuka gradio akan ditampilkan *dialogue box* untuk mengunggah gambar yang akan di prediksi, kemudian akan muncul umpan balik di layar berupa hasil prediksi model.

Kemudian hasil prediksi antara kedua *platform* akan dibandingkan.

7) Diskusi

- Interpretasi Hasil:

Hasil penelitian akan didiskusikan dengan menghubungkannya kembali ke tujuan penelitian dan pertanyaan penelitian. Kelebihan dan kekurangan model CNN dalam mendeteksi cacat kain akan di analisa.

- Perbandingan dengan Penelitian Sebelumnya:

Hasil penelitian akan dibandingkan dengan studi sebelumnya untuk menilai kontribusi dan inovasi yang diberikan. Diskusi akan mencakup analisis bagaimana penggunaan transfer learning mempengaruhi kinerja model.

8) Kesimpulan dan Saran

- Kesimpulan

Kesimpulan akan ditarik berdasarkan hasil analisis dan diskusi, menjawab pertanyaan penelitian utama. Penelitian ini akan menyimpulkan efektivitas penggunaan CNN dan transfer learning dalam mendeteksi dan mengklasifikasikan cacat kain.

- Saran

Saran untuk penelitian lanjutan akan diberikan, termasuk rekomendasi untuk meningkatkan kinerja model dan perluasan jenis cacat yang dapat dideteksi. Saran praktis untuk implementasi model dalam industri tekstil juga akan disampaikan.