

Perancangan Sistem Deteksi dan Pelacakan Objek untuk Pendeteksian PCB di Industri Manufaktur Berbasis Kamera

Syaeful Anas Aklani¹, Nanda Salsabila², Hendi Sama³

^{1,2,3}Sistem Informasi, Universitas Internasional Batam

E-mail: ¹syaeful@uib.ac.id, ²nanda@uib.ac.id, ³hendi@uib.ac.id

Abstrak

Pengembangan sistem inspeksi kualitas pada industri manufaktur, khususnya pada Printed Circuit Board (PCB), menghadapi tantangan signifikan akibat proses manual yang memakan waktu dan berpotensi menimbulkan kesalahan. Penelitian ini bertujuan untuk merancang dan mengembangkan sistem object detection dan tracking berbasis kamera yang mampu mendeteksi serta melacak pergerakan PCB secara otomatis dengan tingkat akurasi tinggi pada kondisi produksi yang dinamis. Metode penelitian yang digunakan adalah research and development (R&D) melalui tahapan analisis kebutuhan, perancangan, implementasi, pengujian, dan evaluasi sistem. Model deteksi dilatih menggunakan algoritma Convolutional Neural Network (CNN) dengan framework YOLO untuk klasifikasi PCB dan Non-PCB, sedangkan pelacakan dilakukan secara real-time. Hasil pengujian menunjukkan bahwa model mencapai akurasi pelatihan dan validasi sebesar 100% dengan nilai precision, recall, dan F1-score sebesar 1.00 pada kedua kelas. Sistem yang dikembangkan terbukti mampu meningkatkan efisiensi proses inspeksi, meminimalkan kesalahan produksi, serta berkontribusi pada upaya transformasi digital di sektor manufaktur Indonesia.

Kata kunci— CNN, object detection, PCB, tracking, YOLO

Abstract

The development of quality inspection systems in manufacturing industries, particularly for Printed Circuit Boards (PCB), faces significant challenges due to time-consuming manual processes and vulnerability to human error. This study aims to design and develop a camera-based object detection and tracking system to automatically identify and track PCB movements with high accuracy under dynamic production conditions. The research employed a Research and Development (R&D) approach consisting of requirement analysis, system design, implementation, testing, and evaluation stages. The detection model was trained using a Convolutional Neural Network (CNN) with the YOLO framework for PCB and Non-PCB classification, while real-time tracking was performed simultaneously. Experimental results show that the model achieved 100% accuracy in both training and validation, with precision, recall, and F1-score values of 1.00 for both classes. The developed system effectively enhances inspection efficiency, minimizes production errors, and contributes to supporting the digital transformation of the manufacturing sector in Indonesia.

Keywords— CNN, object detection, PCB, tracking, YOLO

1. PENDAHULUAN

Industri manufaktur merupakan salah satu sektor strategis yang menjadi tulang punggung perekonomian global. Dalam beberapa dekade terakhir, sektor ini mengalami transformasi besar-besaran seiring dengan hadirnya Revolusi Industri 4.0, yang ditandai oleh integrasi teknologi digital, kecerdasan buatan (*Artificial Intelligence*), dan sistem otomatisasi di berbagai lini produksi. Perkembangan ini tidak hanya mendorong efisiensi, tetapi juga menuntut tingkat akurasi dan kecepatan yang semakin tinggi dalam setiap proses produksi. Salah satu teknologi yang berperan penting dalam mendukung transformasi tersebut adalah *object detection* dan *object tracking*, yaitu kemampuan sistem untuk mengenali, mendeteksi, serta melacak objek secara otomatis melalui citra atau video [1]. Teknologi ini menjadi fondasi penting dalam berbagai aplikasi industri modern, mulai dari sistem keamanan, otomasi, hingga inspeksi kualitas produk.

Dalam konteks pemrosesan citra, kemampuan sistem untuk mengenali objek secara akurat menjadi kunci utama bagi keberhasilan pengambilan keputusan. Akurasi deteksi sangat menentukan apakah sistem dapat membedakan objek yang relevan atau tidak, serta bagaimana sistem tersebut menindaklanjuti hasil deteksi [2]. Sejumlah penelitian sebelumnya telah menunjukkan keberhasilan teknologi *object detection* dalam meningkatkan kecepatan dan efisiensi proses identifikasi objek di berbagai bidang [3]. Namun, penerapannya di sektor industri manufaktur, khususnya manufaktur elektronik, menghadapi tantangan tersendiri karena karakteristik lingkungannya yang kompleks dan dinamis.

Salah satu komponen utama dalam industri manufaktur elektronik adalah *Printed Circuit Board* (PCB). PCB berfungsi sebagai penghubung antar komponen elektronik dan merupakan dasar bagi hampir semua perangkat elektronik modern, mulai dari komputer, ponsel, hingga peralatan industri. Nilai pasar global PCB bahkan diperkirakan mencapai 79 miliar dolar AS pada tahun 2024 [4], menandakan besarnya peran dan kebutuhan terhadap komponen ini. Meskipun demikian, proses inspeksi dan kontrol kualitas PCB di banyak pabrik masih dilakukan secara manual. Metode manual ini sering kali tidak efisien, memakan waktu lama, serta bergantung pada ketelitian manusia yang memiliki keterbatasan dalam hal konsistensi dan kecepatan. Akibatnya, kesalahan dalam inspeksi dapat terjadi, menyebabkan kualitas produk yang tidak seragam, meningkatnya biaya produksi, bahkan potensi kerugian ekonomi bagi perusahaan [5][6].

Selain itu beberapa penelitian yang dilakukan [7] menunjukkan bahwa sistem koreksi posisi berbasis kamera dapat meningkatkan akurasi proses inspeksi pada lini produksi otomatis dengan meminimalkan kesalahan posisi objek sebelum dianalisis. Temuan ini relevan dengan deteksi cacat PCB, karena kualitas citra dan ketepatan posisi komponen PCB merupakan faktor kritis dalam memastikan model vision mengenali pola cacat secara konsisten. Sementara itu, studi [8] menegaskan bahwa transformasi digital memiliki dampak signifikan terhadap peningkatan daya saing industri, termasuk melalui pemanfaatan teknologi cerdas dan otomatisasi pada sektor manufaktur. Integrasi kedua konsep ini dapat mendukung pengembangan sistem deteksi cacat PCB berbasis visi komputer yang tidak hanya akurat secara teknis, tetapi juga memperkuat efisiensi dan daya saing proses produksi melalui pemanfaatan teknologi otomasi dan digitalisasi yang lebih maju.

Permasalahan utama yang muncul dari kondisi tersebut adalah belum adanya sistem otomatis yang mampu mendeteksi dan melacak PCB secara *real-time* dengan tingkat akurasi tinggi, terutama dalam lingkungan produksi yang memiliki variasi pencahayaan, kecepatan gerak objek, dan latar belakang yang kompleks [9]. Di sisi lain, kebutuhan akan sistem inspeksi yang cepat, akurat, dan adaptif semakin mendesak seiring meningkatnya volume dan kompleksitas produksi elektronik. Oleh karena itu, dibutuhkan inovasi dalam bentuk sistem berbasis *computer vision* dan *deep learning* yang dapat menggantikan proses inspeksi manual dengan sistem cerdas berbasis kamera.

Untuk menjawab tantangan ini, penelitian ini menggabungkan dua pendekatan berbasis *deep learning*[10]: algoritma *You Only Look Once* (YOLO) sebagai kerangka deteksi *real-time*, dan model *Convolutional Neural Network* (CNN) khusus sebagai modul deteksi dan klasifikasi lanjutan. YOLO akan berfungsi sebagai tahap awal yang cepat dan efisien untuk lokalisasi PCB (menentukan bounding box dan identitas objek secara *real-time*), sedangkan CNN baik sebagai backbone deteksi yang dipertajam atau sebagai classifier terpisah akan digunakan untuk ekstraksi fitur mendalam dan klasifikasi lebih halus (mis. membedakan PCB vs Non-PCB, serta tahap awal identifikasi potensi cacat atau anomali). Kombinasi ini memanfaatkan keunggulan YOLO dalam kecepatan inferensi dan kemampuan CNN dalam menangkap representasi visual yang kaya, sehingga diharapkan mampu mengatasi variasi kondisi pencahayaan, noise visual, dan bentuk PCB yang berbeda. Selain itu, teknik pendukung seperti *transfer learning*, augmentasi data, dan optimisasi arsitektur CNN akan diterapkan untuk mengurangi kebutuhan data pelatihan besar dan mengoptimalkan kebutuhan komputasi.

Lalu penelitian Pengembangan sistem deteksi PCB dan non-PCB menggunakan algoritma CNN didukung oleh berbagai penelitian yang menekankan kemajuan metode deteksi objek modern. [11] menyoroti evolusi metode deteksi dari CNN hingga transformer, menunjukkan bahwa CNN tetap menjadi fondasi kuat dalam klasifikasi dan segmentasi objek karena kemampuannya mengekstraksi fitur visual secara hierarkis. Dalam konteks PCB, kemampuan ini penting untuk membedakan area berkomponen dari area non-komponen. Kemudian [12] memberikan kontribusi tambahan melalui pemetaan permasalahan bug pada perangkat lunak desain PCB, menegaskan urgensi deteksi kualitas sejak tahap produksi. Studi lain [13] menunjukkan bahwa mekanisme pelacakan berbasis hypergraph matching dapat meningkatkan konsistensi deteksi berulang, yang sangat relevan ketika mengidentifikasi objek PCB pada proses inspeksi otomatis yang melibatkan banyak komponen kecil. Kontribusi lokal seperti penelitian [14] memperlihatkan implementasi machine learning yang efektif untuk mendeteksi kecacatan PCB, memperkuat bukti bahwa CNN mampu memberikan performa tinggi dalam skenario inspeksi visual industri.

Lebih lanjut, penelitian [15] menggarisbawahi efektivitas vision-based component detection dalam memastikan integritas perangkat keras, yang dapat diadaptasi untuk membedakan antara papan PCB yang valid dan objek non-PCB pada lintasan produksi. Kemudian penelitian [16] memperkenalkan YOLOv5_ES yang dioptimalkan untuk mendeteksi cacat kecil pada permukaan PCB, menegaskan bahwa arsitektur berbasis CNN masih sangat relevan untuk mendeteksi objek berukuran kecil dengan presisi tinggi. Studi [17] mengenai pengenalan karakter menggunakan CNN menunjukkan fleksibilitas CNN dalam tugas klasifikasi citra, membuka peluang penerapan serupa untuk membedakan pola spesifik PCB dibandingkan objek lain. [18] juga menekankan bahwa AI berperan penting dalam meningkatkan yield dan mencegah cacat pada manufaktur PCB, yang selaras dengan tujuan deteksi PCB/non-PCB berbasis CNN. Terakhir, metode objek-deteksi ringan seperti SSD dan MobileNetV2 yang diusulkan [19] dapat menjadi inspirasi sistem *real-time* untuk memproses citra PCB dengan cepat, terutama pada lingkungan produksi yang membutuhkan klasifikasi instan antara PCB dan non-PCB.

Berdasarkan pendekatan tersebut, penelitian ini bertujuan merancang dan mengembangkan sistem deteksi dan pelacakan objek berbasis kamera yang mengintegrasikan YOLO dan model CNN untuk mengenali serta melacak keberadaan PCB secara otomatis di lingkungan manufaktur elektronik. Fokus utama adalah memastikan kemampuan kerja secara *real-time*, adaptif terhadap kondisi produksi yang bervariasi, serta menyediakan fondasi yang kuat bagi pengembangan sistem inspeksi kualitas otomatis di masa depan.

Dengan adanya pengembangan ini, diharapkan sistem yang dirancang dapat menjadi solusi terhadap keterbatasan metode inspeksi manual yang masih banyak digunakan di industri saat ini. Selain meningkatkan efisiensi dan akurasi, sistem ini juga berkontribusi dalam mendukung transformasi digital dan penerapan prinsip otomasi cerdas di sektor manufaktur.

Pada akhirnya, penelitian ini diharapkan mampu memperkuat daya saing industri manufaktur elektronik nasional maupun global, sejalan dengan semangat Revolusi Industri 4.0 yang berfokus pada efisiensi, presisi, dan inovasi berkelanjutan.

2. METODE

Penelitian ini menggunakan pendekatan *Research and Development* (R&D) untuk merancang, mengembangkan, dan mengevaluasi sistem deteksi dan pelacakan objek *Printed Circuit Board* (PCB) berbasis kamera. Pendekatan ini dipilih karena fokusnya pada penciptaan solusi praktis yang dapat diterapkan langsung di lingkungan industri.

2.1 Analisis Masalah

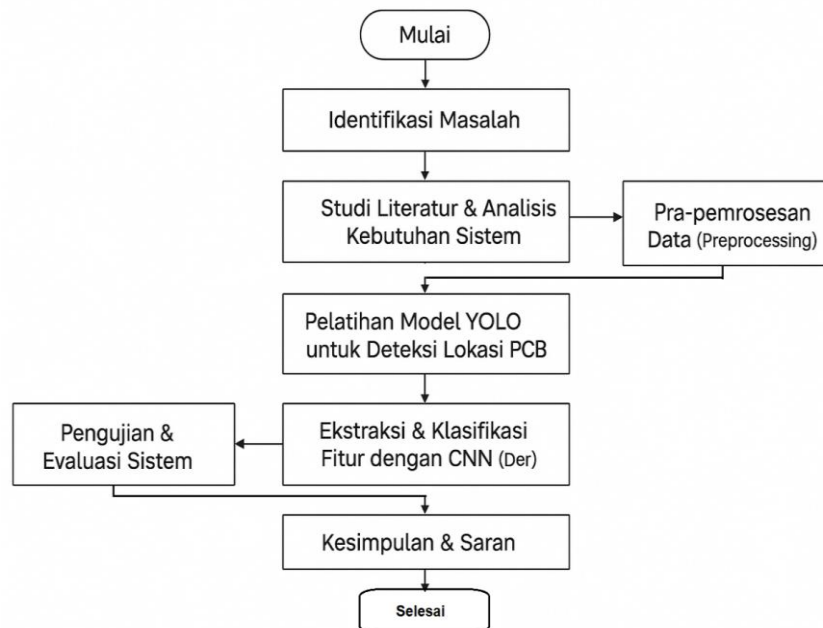
Tahap analisis masalah bertujuan untuk memahami kebutuhan sistem deteksi dan pelacakan PCB di lingkungan produksi. Analisis dilakukan melalui:

1. Studi literatur terkait metode deteksi objek, pelacakan multi-objek, dan sistem pengolahan citra.
2. Observasi langsung di lini produksi untuk mengidentifikasi permasalahan yang dihadapi operator dalam inspeksi PCB.

Hasil analisis ini menjadi dasar untuk perancangan sistem yang sesuai dengan kebutuhan industri.

2.2 Desain Penelitian

Adapun desain penelitian yang dilakukan adalah sebagai mengikuti prosedur standar sesuai dengan sistem deteksi berbasis CNN dan menggunakan model Yolo. Prosedur penelitian adalah langkah-langkah yang dilakukan peneliti untuk mencapai tujuan dari penelitian ini.



Gambar 1. Desain Penelitian

Gambar 1 menggambarkan diagram alir penelitian yang menjelaskan tahapan pengembangan sistem deteksi dan pelacakan objek untuk pendeteksian Printed Circuit Board (PCB) di industri manufaktur berbasis kamera, dengan memanfaatkan algoritma YOLO dan CNN. Proses dimulai dari tahap identifikasi masalah, yaitu mengamati permasalahan utama dalam proses inspeksi PCB yang masih dilakukan secara manual, yang menyebabkan inefisiensi, waktu inspeksi lama, serta potensi kesalahan manusia. Setelah permasalahan diidentifikasi, langkah berikutnya adalah studi literatur dan analisis kebutuhan sistem, di mana dilakukan pengumpulan referensi terkait teknologi object detection, object tracking, serta arsitektur CNN dan YOLO untuk menentukan kebutuhan teknis dan non-teknis sistem yang akan dirancang.

Selanjutnya, dilakukan pra-pemrosesan data (preprocessing), mencakup proses penyiapan dataset gambar PCB yang meliputi normalisasi, augmentasi, dan anotasi bounding box agar data siap digunakan dalam tahap pelatihan model. Tahap berikutnya adalah pelatihan model YOLO untuk mendeteksi posisi dan keberadaan PCB dalam citra atau video secara real-time. Model YOLO dipilih karena kemampuannya mendeteksi objek dengan cepat dan efisien di lingkungan industri yang dinamis. Setelah deteksi posisi PCB dilakukan, tahap ekstraksi dan klasifikasi fitur menggunakan CNN diterapkan untuk mengenali karakteristik spesifik PCB, misalnya pola sirkuit atau keberadaan komponen, sehingga sistem mampu membedakan antara PCB normal dan yang bermasalah.

Kemudian, sistem yang telah dikembangkan diuji pada tahap pengujian dan evaluasi sistem. Evaluasi dilakukan menggunakan beberapa metrik performa utama dalam bidang object detection dan classification, yaitu:

- 1 *Accuracy* (Akurasi): menunjukkan persentase prediksi sistem yang benar dibandingkan dengan seluruh data uji. Akurasi yang tinggi menunjukkan kemampuan sistem dalam mengidentifikasi PCB secara tepat.
- 2 *Precision* (Ketepatan): mengukur seberapa banyak hasil deteksi yang benar-benar PCB dibandingkan seluruh hasil deteksi yang diidentifikasi sebagai PCB. Nilai *precision* yang tinggi menandakan sistem jarang memberikan deteksi palsu (*false positive*).
- 3 *Recall* (Sensitivitas): menunjukkan kemampuan sistem dalam menemukan semua objek PCB yang ada di data uji. Nilai *recall* yang tinggi berarti sistem mampu mendeteksi sebagian besar PCB tanpa terlewat (*false negative* rendah).
- 4 F1-Score: merupakan kombinasi harmonis dari *precision* dan *recall*, memberikan gambaran seimbang antara kemampuan sistem mendeteksi dan menghindari kesalahan.

Tahap terakhir adalah melakukan desain interface dan pada model yang telah dibuat, sehingga dapat diterapkan pada lingkungan real time serta Langkah terakhir Adalah penarikan kesimpulan dan pemberian saran, di mana hasil pengujian dianalisis untuk menilai efektivitas sistem dalam kondisi nyata serta memberikan rekomendasi untuk pengembangan tahap berikutnya, seperti integrasi ke dalam sistem inspeksi kualitas yang lebih kompleks atau penerapan model yang lebih efisien untuk produksi skala besar. Diagram ini secara keseluruhan mencerminkan alur penelitian yang sistematis, mulai dari perumusan masalah hingga evaluasi performa sistem berbasis kamera untuk mendukung otomatisasi di industri manufaktur PCB.

2.3. Spesifikasi Perangkat Keras dan Perangkat Lunak

Sistem dikembangkan dengan memperhatikan kebutuhan perangkat keras dan perangkat lunak agar proses *inferensi machine learning* berjalan efisien. Peralatan yang digunakan ditunjukkan pada Tabel 1 dan Tabel 2.

Tabel 1. *Tool* Penelitian

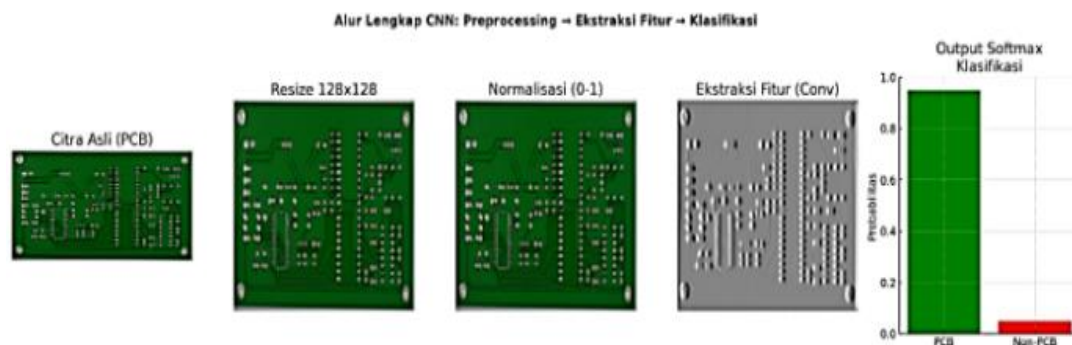
| Komponen | Rekomendasi |
|----------------------|---|
| PC/Laptop | CPU: Intel i5/i7 atau AMD Ryzen 5 ke atas - RAM: 16GB DDR4 ke atas - SSD: 512 GB ke atas |
| GPU (untuk training) | NVIDIA GeForce RTX 3060 / 3070 / 3080 / 3090 (minimal 8GB VRAM) - Atau RTX 4060/4070/4090 (lebih cepat dan Efisien) |
| Kamera | Webcam Logitech (untuk uji awal) - Kamera industri seperti Basler" |
| Sistem Operasi | Windows 10/11, Linux (Ubuntu 20.04/22.04), atau WSL 2 (jika di Windows) |

Tabel 2. *Tool* Perangkat Lunak

| Software | Versi | Fungsi |
|--------------|------------------------------|--|
| OS | Windows 10/11 | Untuk pengembangan awal, kompatibel dengan CUDA dan tools AI |
| Python | 3.8 – 3.11 | Bahasa utama untuk pengembangan model |
| PyTorch | ≥ 1.10 (dengan CUDA support) | Framework utama untuk YOLO (YOLO v5/v7/v8 menggunakan PyTorch) |
| TorchVision | Sesuai versi PyTorch | Untuk preprocessing gambar |
| CUDA & cuDNN | CUDA 11.8 / 12.1, cuDNN 8.x | Untuk akselerasi GPU NVIDIA selama training |

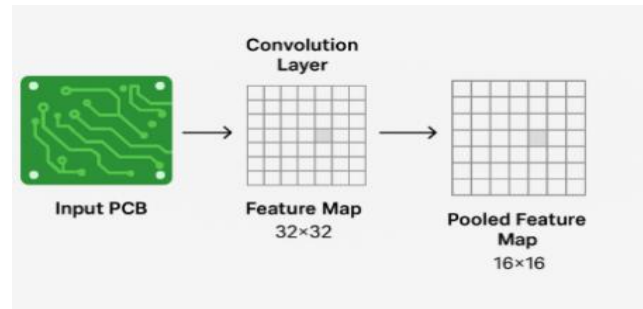
2.4 Arsitektur Model CNN

Model *Convolutional Neural Network* (CNN) digunakan untuk mengekstraksi fitur visual dari citra PCB. Arsitektur CNN terdiri atas beberapa lapisan utama: *convolution*, *pooling*, dan *fully connected layer*, seperti ditunjukkan pada gambar 2.



Gambar 2. Alur CNN

Lapisan *convolution* mengekstraksi fitur penting dari citra seperti jalur tembaga dan bentuk komponen elektronik menggunakan filter 3×3 dan fungsi aktivasi ReLU (*Rectified Linear Unit*). Selanjutnya, *pooling layer* (menggunakan *max pooling*) mengurangi ukuran data tanpa kehilangan informasi penting, mempercepat proses training, dan mengurangi risiko *overfitting*. Hasil dari beberapa lapisan *convolution* dan *pooling* dimasukkan ke dalam *fully connected layer* untuk klasifikasi akhir menjadi dua kelas: PCB dan Non-PCB. Hal ini divisualisasikan pada Gambar 3.

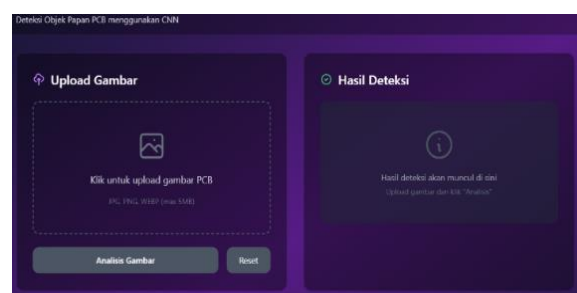
Gambar 3. *Convolution Layer*

Proses pelatihan model dilakukan dengan menggunakan *optimizer* Adam, yang dipilih karena kemampuannya dalam menyesuaikan *learning rate* secara adaptif sehingga mempercepat proses konvergensi model. Untuk fungsi kerugian digunakan *Cross-Entropy Loss*, yang sesuai dengan karakteristik permasalahan klasifikasi biner antara citra PCB dan non-PCB. Setiap lapisan konvolusi pada model menggunakan fungsi aktivasi *Rectified Linear Unit* (ReLU) untuk meningkatkan non-linearitas dan mempercepat proses pelatihan tanpa menimbulkan masalah *vanishing gradient*. Secara keseluruhan, model yang dikembangkan memiliki sebanyak 3.211.779 parameter, yang menunjukkan tingkat kompleksitas arsitektur yang cukup tinggi namun tetap efisien untuk proses inferensi secara *real-time*.

2.5 Implementasi

Implementasi dilakukan berdasarkan desain arsitektur sistem dan konfigurasi perangkat keras serta perangkat lunak yang telah dijelaskan sebelumnya. Pada tahap implementasi, dilakukan pengumpulan dan pelabelan *dataset* gambar PCB yang digunakan untuk melatih model deteksi dan pelacakan objek. Model YOLO dilatih untuk mengenali bentuk dan posisi PCB, sedangkan algoritma MOT digunakan untuk melacak pergerakan objek pada setiap *frame* video. Perangkat keras utama yang digunakan adalah kamera industri beresolusi tinggi dan komputer dengan GPU untuk mempercepat proses pelatihan.

Sistem juga dikembangkan dengan antarmuka pengguna yang sederhana dan interaktif, sehingga pengguna dapat mengunggah gambar PCB dan langsung melihat hasil deteksi secara *real-time*, seperti ditunjukkan pada Gambar 4.



Gambar 4. Antarmuka Sistem

2.6 Pengujian dan Evaluasi

Pengujian dilakukan untuk menilai performa sistem dalam mendeteksi dan melacak PCB di lingkungan produksi yang dinamis. Evaluasi mencakup dua pendekatan:

1. Kuantitatif, dilakukan menggunakan metrik standar deteksi objek, yaitu akurasi (rasio prediksi benar), presisi (*true positive* terhadap prediksi positif), *recall* (*true positive* terhadap data positif), F1-score (harmonik rata-rata presisi dan *recall*), serta kecepatan deteksi (FPS atau *latency*).

2. Kualitatif, untuk menilai kesesuaian sistem dengan kebutuhan industri manufaktur berdasarkan observasi lapangan.

Performa model diukur menggunakan persamaan sebagai berikut:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

Keterangan:

TP (*True Positive*) = jumlah citra PCB yang berhasil dideteksi dengan benar.

TN (*True Negative*) = jumlah citra Non-PCB yang diklasifikasikan dengan benar.

FP (*False Positive*) = jumlah citra Non-PCB yang salah diklasifikasikan sebagai PCB.

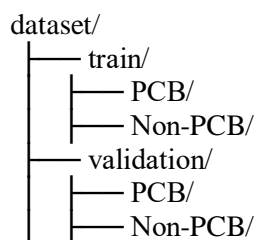
FN (*False Negative*) = jumlah citra PCB yang gagal terdeteksi.

Metrik-metrik ini digunakan untuk mengukur kinerja model dalam mengenali objek dengan tepat, serta menilai keseimbangan antara kemampuan model mendeteksi (*recall*) dan menghindari kesalahan deteksi (*precision*). Hasil pengujian digunakan sebagai dasar untuk melakukan penyempurnaan sistem sehingga dapat bekerja secara *real-time*, adaptif terhadap kondisi pencahayaan, serta efisien dalam mendukung inspeksi PCB otomatis.

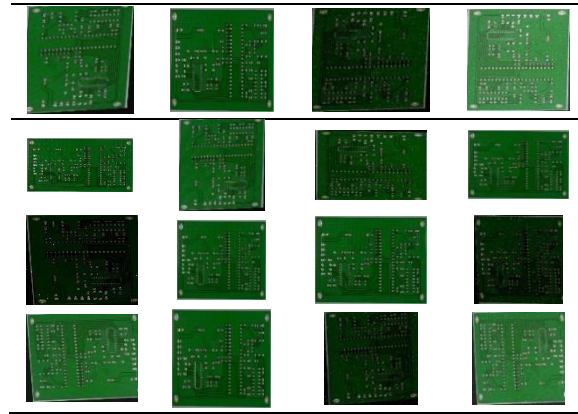
3. HASIL DAN PEMBAHASAN

Sistem deteksi objek PCB ini dikembangkan untuk membantu proses identifikasi komponen elektronik secara otomatis. Antarmuka sistem dirancang sederhana dan interaktif agar pengguna dapat mengunggah gambar PCB dan melihat hasil deteksi secara instan.

Dataset yang digunakan bersumber dari Roboflow dengan total 2.496 gambar, terdiri dari dua kelas yaitu PCB dan Non-PCB. Dataset dibagi menjadi dua bagian, yaitu data latih dan data validasi. Struktur data disusun sebagai berikut:



Dataset latih digunakan untuk proses pembelajaran model, sedangkan dataset validasi digunakan untuk menguji kemampuan model mengenali data baru. Contoh data yang digunakan ditunjukkan pada Tabel 3 (dataset PCB) dan Tabel 4 (dataset Non-PCB). Total dataset yang digunakan adalah 2.496 gambar dari Roboflow, dibagi menjadi kelas PCB dan Non-PCB untuk pelatihan dan validasi. Tahapan pengujian dan analisis hasil ini dilakukan berdasarkan prosedur yang telah dijelaskan pada bagian metode, khususnya pada bagian implementasi dan evaluasi, untuk memastikan kesesuaian antara rancangan metode dan hasil yang diperoleh.



Gambar 5. Dataset PCB



Gambar 6. Dataset Non PCB

Gambar 5 dan 6 menunjukkan contoh dataset yang digunakan untuk pelatihan model. Model *Convolutional Neural Network* (CNN) dikembangkan dengan arsitektur yang terdiri dari lapisan konvolusi, pooling, dan fully connected untuk mengekstraksi fitur visual. Proses analisis hasil dilakukan melalui beberapa tahapan yang mengacu pada metode penelitian. Pertama, model CNN yang telah dilatih menggunakan data latih dijalankan untuk memproses dataset validasi guna mengukur akurasi dan *loss* pada tiap *epoch*. Kedua, hasil inferensi model dibandingkan dengan label sebenarnya (*ground truth*) untuk menghitung nilai akurasi, presisi, *recall*, dan *F1-score*. Ketiga, model diuji pada data uji terpisah untuk memastikan kemampuan generalisasi terhadap citra baru. Dari hasil tersebut, dilakukan analisis tren *training* dan *validation loss* untuk menilai stabilitas model serta mendeteksi kemungkinan *overfitting*.

Hasil pelatihan menunjukkan konvergensi yang cepat, dengan akurasi pelatihan mencapai 100% pada *epoch* ke-4. Namun, akurasi validasi menunjukkan fluktuasi, yang mengindikasikan adanya *overfitting* sementara akibat ukuran data validasi yang kecil seperti terlihat pada Tabel 3.

Tabel 3. *Training dan Testing Dataset*

| <i>Epoch</i> | <i>Training Accuracy</i> | <i>Training Loss</i> | <i>Validation Accuracy</i> | <i>Validation Loss</i> |
|--------------|--------------------------|----------------------|----------------------------|------------------------|
| 1 | 0.344 | 0.729 | 0.750 | 0.732 |
| 2 | 0.800 | 0.386 | 0.750 | 0.426 |
| 3 | 0.688 | 0.356 | 0.250 | 1.117 |
| 4 | 1.000 | 0.381 | 0.375 | 0.812 |
| 5 | 0.800 | 0.499 | 0.875 | 0.386 |
| 6 | 0.938 | 0.211 | 0.875 | 0.550 |

| Epoch | Training Accuracy | Training Loss | Validation Accuracy | Validation Loss |
|-------|-------------------|---------------|---------------------|-----------------|
| 7 | 0.906 | 0.267 | 0.875 | 0.265 |
| 8 | 1.000 | 0.071 | 1.000 | 0.109 |
| 9 | 1.000 | 0.013 | 0.625 | 0.380 |
| 10 | 1.000 | 0.026 | 0.500 | 0.750 |
| 11 | 1.000 | 0.017 | 0.500 | 0.605 |
| 12 | 1.000 | 0.037 | 1.000 | 0.089 |
| 13 | 1.000 | 0.004 | 1.000 | 0.043 |
| 14 | 1.000 | 0.001 | 0.875 | 0.173 |
| 15 | 1.000 | 0.000 | 0.875 | 0.402 |
| 16 | 1.000 | 0.021 | 0.875 | 0.419 |
| 17 | 1.000 | 0.000 | 0.875 | 0.427 |
| 18 | 1.000 | 0.012 | 0.875 | 0.274 |
| 19 | 1.000 | 0.000 | 0.875 | 0.144 |
| 20 | 1.000 | 0.000 | 1.000 | 0.057 |

Meskipun demikian, model akhirnya menunjukkan performa optimal dengan validation loss yang konsisten menurun yang diperlihatkan pada Tabel 4 dan Tabel 5.

Tabel 4. Data Uji PCB dan Non-PCB

| Actual/Predicted | Kelas 0 (non-PCB) | Kelas 1 (PCB) |
|-------------------|-------------------|---------------|
| Kelas 0 (Non-PCB) | 6 | 0 |
| Kelas 1 (PCB) | 0 | 2 |

Tabel 5. Laporan Klasifikasi

| Kelas | Precision | Recall | F1-Score | Support |
|---------------|-----------|--------|----------|---------|
| 0.0 (Non-PCB) | 1.00 | 1.00 | 1.00 | 6 |
| 1.0 (PCB) | 1.00 | 1.00 | 1.00 | 2 |
| Accuracy | | | 1.00 | 8 |
| Macro Avg | 1.00 | 1.00 | 1.00 | 8 |
| Weighted Avg | 1.00 | 1.00 | 1.00 | 8 |

Setelah pelatihan, model dievaluasi menggunakan data uji terpisah. *Confusion matrix* dan *classification report* menunjukkan performa sempurna dengan *precision*, *recall*, dan *F1-score* 100% untuk kedua kelas (PCB dan Non-PCB) yang dapat dilihat pada Gambar 7. Hasil ini menegaskan kemampuan model dalam membedakan kedua kelas tersebut secara akurat pada dataset uji yang digunakan.

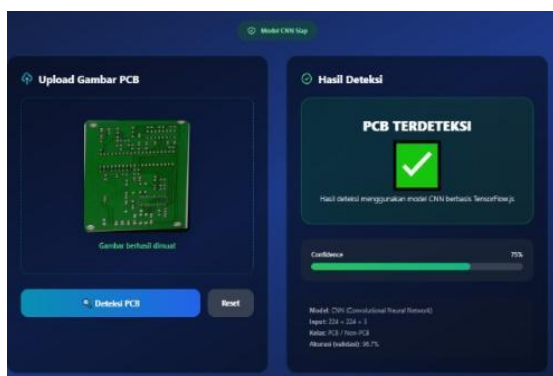
| Layer (type) | Output Shape | Param # |
|--------------------------------|----------------------|-----------|
| conv2d (Conv2D) | (None, 126, 126, 32) | 896 |
| max_pooling2d (MaxPooling2D) | (None, 63, 63, 32) | 0 |
| conv2d_1 (Conv2D) | (None, 61, 61, 64) | 18,496 |
| max_pooling2d_1 (MaxPooling2D) | (None, 30, 30, 64) | 0 |
| conv2d_2 (Conv2D) | (None, 28, 28, 128) | 73,856 |
| max_pooling2d_2 (MaxPooling2D) | (None, 14, 14, 128) | 0 |
| flatten (Flatten) | (None, 25088) | 0 |
| dense (Dense) | (None, 128) | 3,211,392 |
| dense_1 (Dense) | (None, 1) | 129 |

Gambar 7. Hasil Training

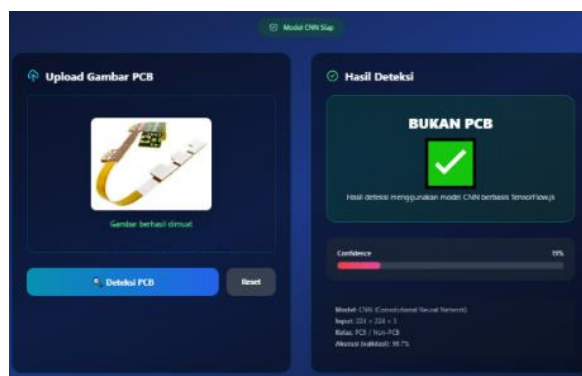
Gambar 7 menjelaskan Arsitektur model CNN yang digunakan dalam penelitian ini terdiri dari beberapa lapisan utama yang bekerja secara berurutan untuk mendeteksi dan mengklasifikasikan objek PCB secara efisien. Model ini memiliki arsitektur efektif dengan total 3.211.779 parameter. Lapisan Conv2D pertama, ketiga, dan kelima berfungsi untuk mengekstraksi berbagai fitur penting dari citra, seperti tepi, pola, maupun bentuk komponen PCB. Jumlah filter pada lapisan ini meningkat secara bertahap dari 32, 64, hingga 128, yang menandakan bahwa jaringan melakukan pembelajaran fitur dari tingkat yang sederhana hingga kompleks. Selanjutnya, setiap lapisan MaxPooling2D yang menyertainya (lapisan ke-2, ke-4, dan ke-6) berperan dalam mengurangi dimensi spasial citra, sehingga proses komputasi menjadi lebih efisien dan risiko overfitting dapat diminimalkan, tanpa kehilangan informasi penting dari fitur yang telah diekstraksi.

Setelah proses ekstraksi fitur selesai, hasil keluaran dari lapisan konvolusi diubah menjadi bentuk vektor satu dimensi melalui lapisan Flatten (lapisan ke-7). Tahapan ini penting agar data fitur yang telah diekstraksi dapat diolah oleh lapisan berikutnya, yaitu lapisan Dense. Pada lapisan Dense pertama (lapisan ke-8) yang memiliki 128 neuron, dilakukan proses klasifikasi awal terhadap fitur-fitur PCB yang telah diekstraksi. Sementara itu, lapisan Dense terakhir (lapisan ke-9) berfungsi sebagai output layer dengan satu neuron yang menghasilkan nilai prediksi akhir, misalnya 1 untuk objek PCB dan 0 untuk non-PCB. Kombinasi keseluruhan lapisan ini memungkinkan model CNN untuk mengenali dan membedakan objek PCB secara akurat melalui proses pembelajaran mendalam terhadap citra masukan.

Sedangkan hasil deploy model menunjukkan kemampuan sistem dalam beroperasi secara praktis. Saat gambar PCB diunggah, sistem berhasil mendeteksinya dengan tingkat kepercayaan 75% seperti diperlihatkan pada Gambar 8. Demikian pula, sistem berhasil mengidentifikasi objek *non-PCB* dengan tingkat kepercayaan rendah, menunjukkan kemampuannya untuk membedakan objek yang bukan merupakan *Printed Circuit Board* sebagaimana terlihat pada Gambar 9.



Gambar 8. Hasil Deteksi PCB



Gambar 9. Hasil Deteksi Non-PCB

Secara keseluruhan, hasil penelitian menunjukkan bahwa metode CNN berbasis YOLOv8 efektif dalam mendeteksi citra PCB. Kombinasi *optimizer* Adam, fungsi aktivasi ReLU, dan *Cross-Entropy Loss* memberikan proses pelatihan yang stabil dan cepat konvergen. Namun demikian, keterbatasan jumlah data dan ukuran sampel uji yang kecil perlu mendapat perhatian pada pengujian lanjutan agar kemampuan generalisasi model dapat dievaluasi lebih komprehensif.

4. KESIMPULAN

Berdasarkan hasil penelitian mengenai perancangan sistem *object detection* dan *tracking* untuk mendeteksi objek PCB pada industri manufaktur berbasis kamera, dapat disimpulkan bahwa penelitian ini berhasil merancang dan mengimplementasikan sistem deteksi otomatis menggunakan algoritma CNN dengan dukungan *framework* YOLO yang mampu mengklasifikasikan PCB dan Non-PCB dengan tingkat akurasi sangat tinggi. Hasil pelatihan menunjukkan akurasi model mencapai 100% sejak *epoch* ke-4, meskipun akurasi validasi sempat fluktuatif akibat keterbatasan jumlah data uji, namun pada akhirnya mampu mencapai akurasi validasi 100% dengan *validation loss* yang sangat rendah. Berdasarkan pengujian menggunakan *confusion matrix*, diperoleh nilai *precision*, *recall*, dan *F1-score* sebesar 1.00 pada kedua kelas, yang menegaskan kemampuan sistem dalam membedakan objek secara sempurna. Antarmuka sistem yang dikembangkan juga terbukti memudahkan pengguna dalam melakukan *upload* gambar, analisis deteksi, serta melihat hasil klasifikasi secara cepat dan interaktif tanpa perlu *preprocessing* manual. Meskipun performa sistem menunjukkan hasil yang optimal, penelitian ini memiliki keterbatasan berupa jumlah dataset yang masih terbatas sehingga berpotensi mempengaruhi kemampuan generalisasi model pada kondisi nyata dengan variasi data yang lebih kompleks. Oleh karena itu, penelitian selanjutnya disarankan untuk menggunakan dataset yang lebih besar dan seimbang, menerapkan teknik pencegahan *overfitting* seperti *dropout*, *early stopping*, dan *data augmentation*, serta melakukan pengujian langsung pada lingkungan industri nyata dengan kondisi pencahayaan dan kecepatan produksi yang bervariasi. Selain itu, sistem dapat dikembangkan lebih lanjut dengan menambahkan kemampuan deteksi cacat spesifik pada PCB serta diintegrasikan dengan teknologi IoT dan *cloud computing* agar hasil deteksi dapat dipantau secara *real-time*, mendukung transformasi menuju industri 4.0.

REFERENSI

- [1] B. Mirzaei, H. Nezamabadi-pour, A. Raoof, and R. Derakhshani, "Small Object Detection and Tracking: A Comprehensive Review," *Sensors*, vol. 23, no. 15, p. 6887, Aug. 2023, <https://doi.org/10.3390/s23156887>
- [2] M. Shen et al., "Defect detection of printed circuit board assembly based on YOLOv5," *Scientific Reports*, vol. 14, no. 1, p. 19287, Aug. 2024, <https://doi.org/10.1038/s41598-024-70176-1>
- [3] T. Saidani, R. Ghodhbani, A. Alhomoud, A. Alshammari, H. Zayani, and M. ben Ammar, "Hardware Acceleration for Object Detection using YOLOv5 Deep Learning Algorithm on Xilinx Zynq FPGA Platform," *Engineering, Technology & Applied Science Research*, vol. 14, no. 1, pp. 13066–13071, Feb. 2024, <https://doi.org/10.48084/etasr.6761>
- [4] Ixiaohuihuihui, "Tiny-Defect-Detection-for-PCB," GitHub repository. <https://github.com/Ixiaohuihuihui/Tiny-Defect-Detection-for-PCB>
- [5] I.-C. Chen, R.-C. Hwang, and H.-C. Huang, "PCB Defect Detection Based on Deep Learning Algorithm," *Processes*, vol. 11, no. 3, p. 775, Mar. 2023, <https://doi.org/10.3390/pr11030775>
- [6] L. Cai and J. Li, "PCB defect detection system based on image processing," *Journal of Physics: Conference Series*, vol. 2383, no. 1, p. 012077, Dec. 2022, <https://doi.org/10.1088/1742-6596/2383/1/012077>
- [7] A. K. Bedaka, S.-C. Lee, A. M. Mahmoud, Y.-S. Cheng, and C.-Y. Lin, "A Camera-Based Position Correction System for Autonomous Production Line Inspection," *Sensors*, vol. 21, no. 12, p. 4071, Jun. 2021, <https://doi.org/10.3390/s21124071>

- [8] H. Kurnianingsih, "The Impact of Digital Transformation on Corporate Competitiveness in Indonesia's Creative Economy Sector," *Journal of the American Institute*, vol. 2, no. 5, pp. 648–657, May 2025, <https://doi.org/10.71364/a6370533>
- [9] J. A. Rodríguez-Rodríguez, E. López-Rubio, J. A. Ángel-Ruiz, and M. A. Molina-Cabello, "The Impact of Noise and Brightness on Object Detection Methods," *Sensors*, vol. 24, no. 3, p. 821, Jan. 2024, <https://doi.org/10.3390/s24030821>
- [10] S. S. A. Zaidi, M. S. Ansari, A. Aslam, N. Kanwal, M. Asghar, and B. Lee, "A survey of modern deep learning based object detection models," *Digital Signal Processing*, vol. 126, p. 103514, Jun. 2022, <https://doi.org/10.1016/j.dsp.2022.103514>
- [11] E. Arkin, N. Yadikar, X. Xu, A. Aysa, and K. Ubul, "A survey: object detection methods from CNN to transformer," *Multimedia Tools and Applications*, vol. 82, no. 14, pp. 21353–21383, Jun. 2023, <https://doi.org/10.1007/s11042-022-13801-3>
- [12] X. Zhao et al., "A Comprehensive Study of Open-Source Printed Circuit Board (PCB) Design Software Bugs," *IEEE Transactions on Instrumentation and Measurement*, vol. 73, pp. 1–16, 2024, <https://doi.org/10.1109/TIM.2024.3450918>
- [13] Z. Cui, Y. Dai, Y. Duan, and X. Tao, "Joint Object Detection and Multi-Object Tracking Based on Hypergraph Matching," *Applied Sciences*, vol. 14, no. 23, p. 11098, Nov. 2024, <https://doi.org/10.3390/app142311098>
- [14] A. Zainul Muttaqin, I. Nawawi, and J. Pamungkas, "Implementasi Machine Learning pada Deteksi Kecacatan Printed Circuit Board," *Voteteknika (Vocational Teknik Elektronika dan Informatika)*, vol. 12, no. 1, p. 44, Mar. 2024, <https://doi.org/10.24036/voteteknika.v12i1.127224>
- [15] W. Zhao, S. R. Gurudu, S. Taheri, S. Ghosh, M. A. Mallaiyan Sathiaselvan, and N. Asadizanjani, "PCB Component Detection Using Computer Vision for Hardware Assurance," *Big Data and Cognitive Computing*, vol. 6, no. 2, p. 39, Apr. 2022, <https://doi.org/10.3390/bdcc6020039>
- [16] Y. Gao, Z. Li, Y. Wang, and S. Zhu, "A Novel YOLOv5_ES based on lightweight small object detection head for PCB surface defect detection," *Scientific Reports*, vol. 14, no. 1, p. 23650, Oct. 2024, <https://doi.org/10.1038/s41598-024-74368-7>
- [17] M. Mustafa, N. Nur, F. Wajidi, A. A. A. Cirua, and Ismaun, "Pengenalan Huruf Aksara Lontara Menggunakan Metode Convolutional Neural Network," *Journal of Computer and Information System (J-CIS)*, vol. 7, no. 1, pp. 34–42, Mar. 2024, <https://doi.org/10.31605/jcis.v7i1>
- [18] H. Ghelani, "Advanced AI Technologies for Defect Prevention and Yield Optimization in PCB Manufacturing," *International Journal of Engineering and Computer Science*, vol. 13, no. 10, pp. 26534–26550, Oct. 2024, <https://doi.org/10.18535/ijecs/v13i10.4924>
- [19] A. Purnama, J. Indra, S. Arum Puspita Lestari, and S. Faisal, "Deteksi Pelanggaran Penggunaan Helm Dengan Metode Ssd Dan Arsitektur Mobilenetv2," *Journal of Information System Management (JOISM)*, vol. 7, no. 1, pp. 7–14, Jun. 2025, <https://doi.org/10.24076/joism.2025v7i1.2071>