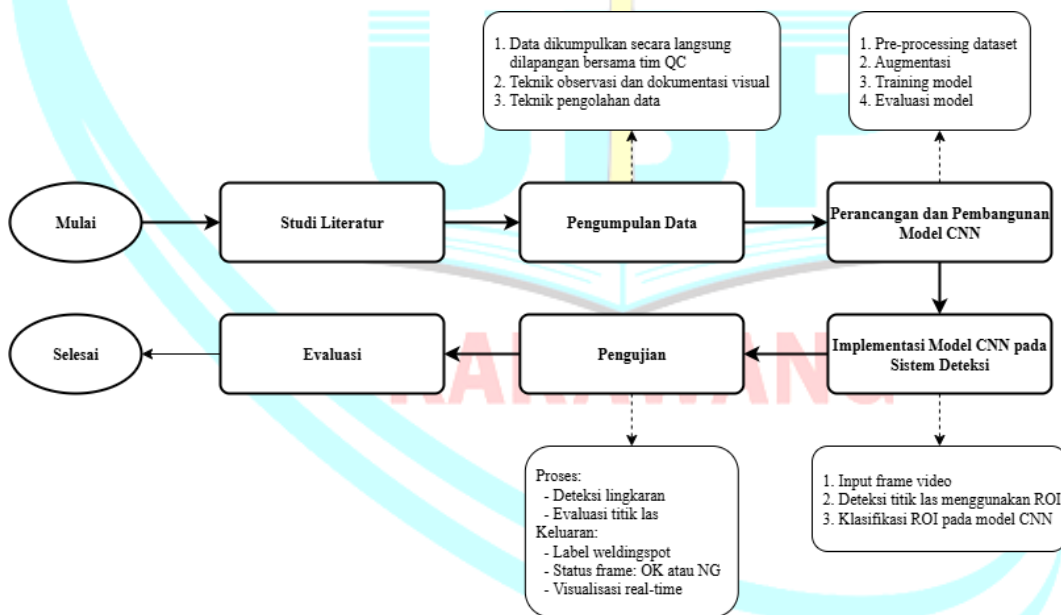


BAB III METODE PENELITIAN

3.1 Objek Penelitian

Objek dalam penelitian ini adalah citra hasil *spot welding* pada area *rear door*, khususnya pada bagian *beam protect* kendaraan. Citra yang digunakan berasal dari proses produksi aktual dan telah dikategorikan ke dalam dua kelas, yaitu *weldingspot* (las yang sesuai) dan *notweldspot* (las cacat atau tidak sesuai). Data ini digunakan untuk pelatihan dan pengujian model klasifikasi berbasis *Convolutional Neural Network* (CNN).

3.2 Prosedur Penelitian



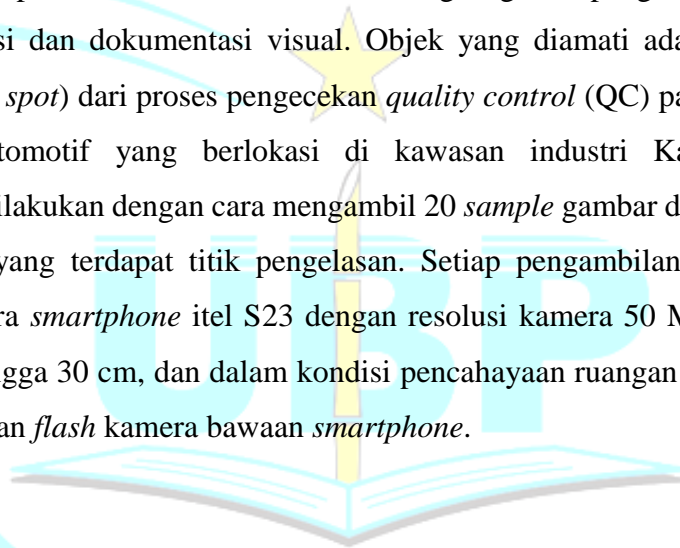
Gambar 3. 1 Prosedur Penelitian

Gambar 3. 1 menunjukkan alur prosedur penelitian yang terdiri dari beberapa tahapan utama. Penelitian diawali dengan studi literatur sebagai dasar teori, kemudian dilanjutkan dengan pengumpulan data yang dilakukan secara langsung di lapangan bersama tim QC menggunakan teknik observasi dan dokumentasi visual. Setelah data terkumpul, dilakukan perancangan dan pembangunan model CNN yang mencakup proses *pre-processing*, augmentasi data, pelatihan model, dan evaluasi awal. Model yang telah dibangun kemudian diimplementasikan pada

sistem deteksi visual berbasis video, yang mencakup input video, deteksi titik las dengan ROI, serta klasifikasi menggunakan CNN. Tahap pengujian dilakukan untuk mengevaluasi sistem dengan memproses *frame video* guna mendeteksi lingkaran dan mengevaluasi titik las. Hasil keluaran sistem berupa label klasifikasi, status *frame* (OK/NG), dan visualisasi secara *real-time*. Prosedur diakhiri dengan evaluasi, yang bertujuan untuk menilai performa keseluruhan sistem.

3.3 Pengumpulan Data

Teknik pengumpulan data dilakukan secara langsung di lapangan melalui pendekatan observasi dan dokumentasi visual. Objek yang diamati adalah titik pengelasan (*welding spot*) dari proses pengecekan *quality control* (QC) pada salah satu perusahaan otomotif yang berlokasi di kawasan industri Karawang. Pengumpulan data dilakukan dengan cara mengambil 20 *sample* gambar dari objek area *beam protect* yang terdapat titik pengelasan. Setiap pengambilan gambar menggunakan kamera *smartphone* itel S23 dengan resolusi kamera 50 MP, pada jarak berkisar 20 hingga 30 cm, dan dalam kondisi pencahayaan ruangan 250 *Lux* yang didukung dengan *flash* kamera bawaan *smartphone*.

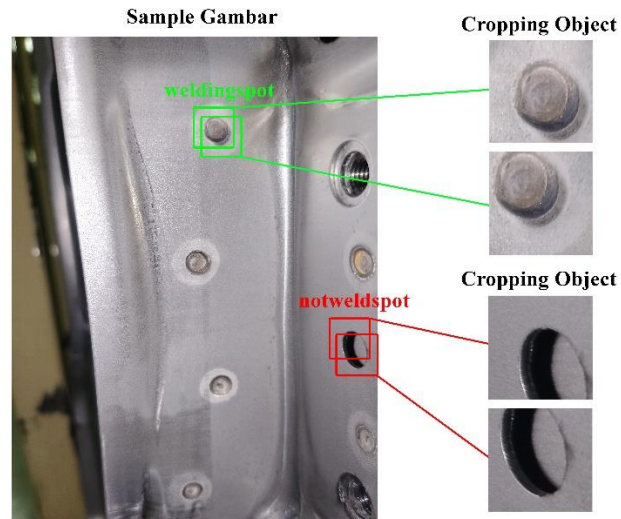


KARAWANG



Gambar 3. 2 *Sample Gambar Area Beam Protect*

Kegiatan dokumentasi ini dilakukan bersama tim QC perusahaan guna memastikan bahwa setiap foto yang diambil benar-benar mewakili kondisi aktual hasil las yang telah diperiksa secara manual. Setelah pengambilan gambar, dilakukan proses akuisisi data dengan teknik *cropping* secara manual terhadap masing-masing gambar. *Cropping* dilakukan untuk setiap objek titik las dalam gambar, guna memperoleh citra yang hanya berfokus pada area penting, yaitu titik pengelasan (*welding spot*) maupun bukan titik pengelasan (*not welding spot*). Untuk memperkaya variasi data, *cropping* dilakukan dengan berbagai variasi pada setiap objek. Variasi tersebut mencakup perbedaan dalam ukuran area *cropping* serta posisi atau sudut *cropping*. Tujuan dari variasi *cropping* ini adalah untuk meningkatkan keragaman *dataset*, sehingga model dapat mengenali berbagai bentuk dan kondisi visual dari titik las, serta menghindari *overfitting* terhadap pola tertentu.



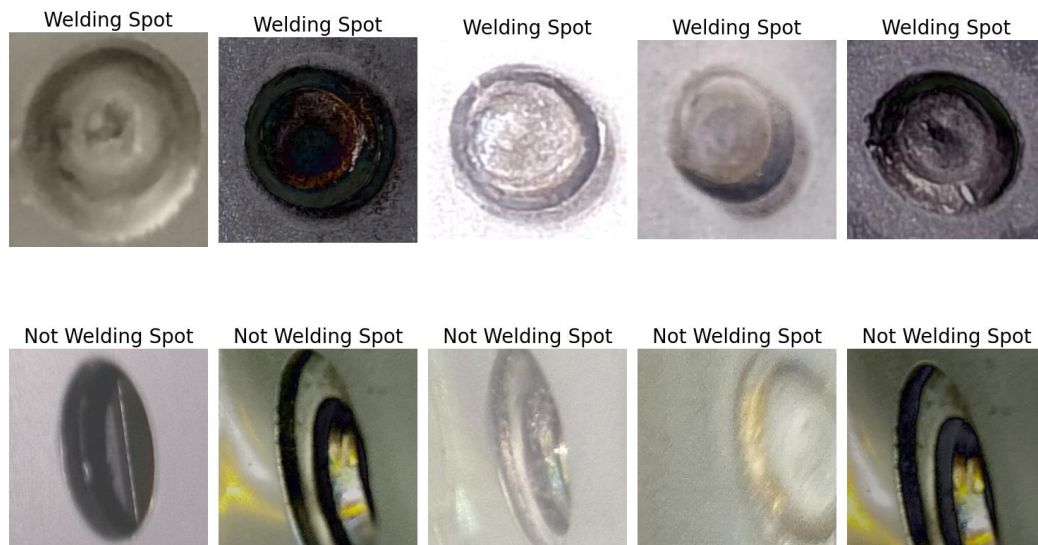
Gambar 3. 3 Teknik Akuisisi Citra

Setelah melalui proses akuisisi, data dibagi menjadi 2 kelas, yaitu *welding spot* yang menunjukkan bahwa pengelasan telah dilakukan dengan baik serta *not welding spot* yang menunjukkan objek tidak identik atau bukan objek *spot* pengelasan. Seluruh gambar disimpan dalam format JPG dan diorganisasi ke dalam dua folder terpisah berdasarkan kategorinya. Gambar-gambar ini nantinya digunakan sebagai *dataset* pelatihan dan pengujian model klasifikasi citra berbasis *deep learning*, dengan tujuan untuk membangun sistem deteksi otomatis titik las.

Dataset dibagi ke dalam beberapa subset untuk keperluan pelatihan (*training*), validasi (*validation*), dan pengujian (*testing*) model. *Dataset* terdiri dari 540 citra digital yang terbagi ke dalam 2 kelas *weldingspot* dan *notweldspot* dengan pembagian jumlah data yang seimbang agar tidak terjadi bias dalam proses pelatihan model. Proses pembagian data dilakukan menggunakan bantuan pustaka dari *TensorFlow* dan *NumPy*. Data diacak secara acak (*random shuffle*) agar distribusi data dari setiap kelas tetap merata pada masing-masing subset.

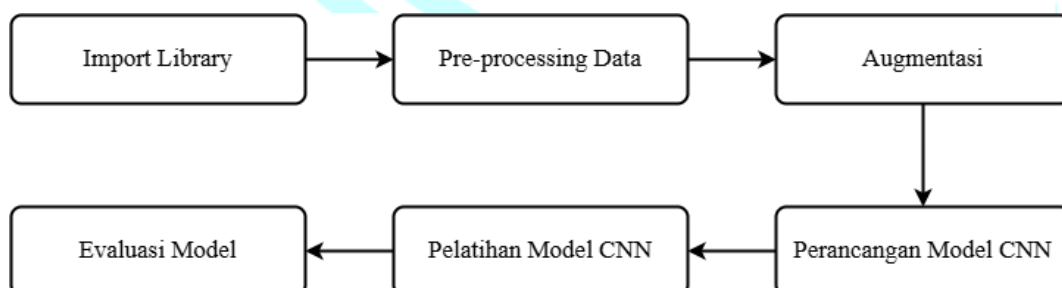
Tabel 3. 1 Rincian *Dataset*

Kelas	<i>Train_data</i>	<i>Val_data</i>	<i>Test_data</i>	Jumlah
<i>weldingspot</i>	189	54	27	270
<i>notweldspot</i>	189	54	27	270

Gambar 3. 4 Dokumentasi *Dataset*

3.4 Perancangan dan Pembangunan Model CNN

Untuk dapat membuat sebuah sistem yang dapat mendeteksi *welding spot* melalui citra, diperlukan suatu model yang dapat dibangun untuk mengklasifikasikan antara citra yang mengandung objek *welding spot* dan citra yang bukan atau tidak terdapat objek *welding spot*. Berikut alur pembangunan model menggunakan algoritma CNN yang akan diimplementasikan pada sistem deteksi citra dan dilakukan menggunakan perangkat lunak *Google Colaboratory*, tersaji pada gambar dibawah ini.

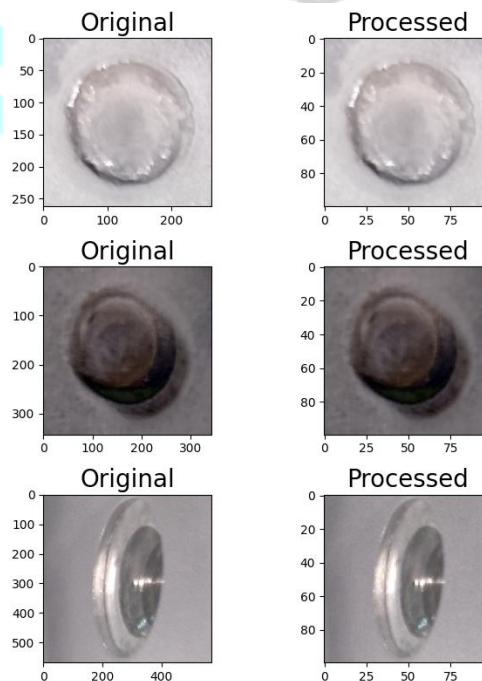


Gambar 3. 5 Alur Perancangan Sistem Pembangunan Model

Keterangan :

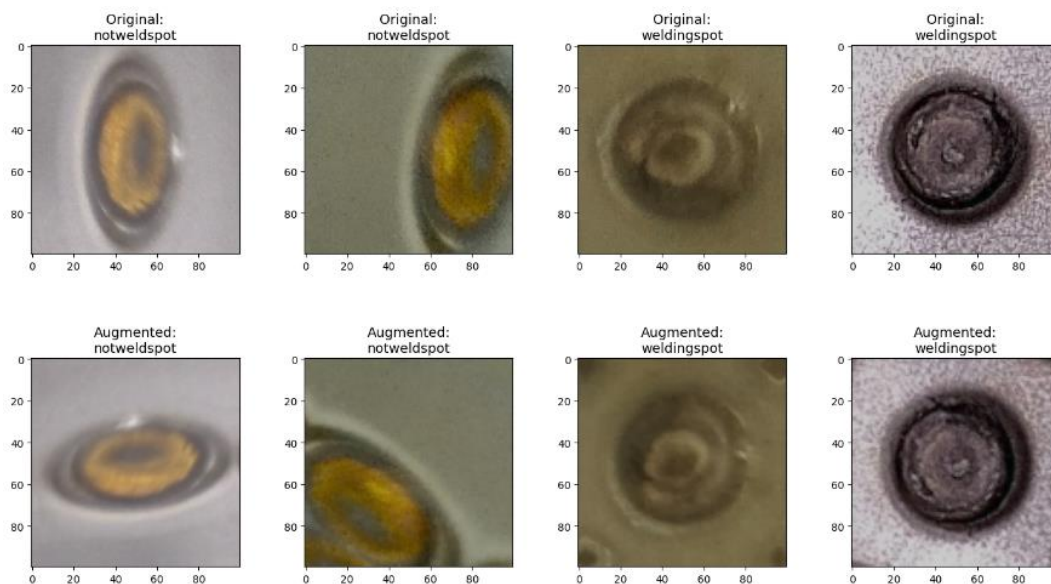
1. Pada proses awal dilakukan tahapan import pustaka *python* diantaranya:
 - a. *TensorFlow* dan *Keras* digunakan sebagai kerangka kerja (*framework*) utama dalam membangun, melatih, dan menguji model *Convolutional Neural Network* (CNN). Modul yang digunakan meliputi *tensorflow.keras.layers*

- untuk membangun arsitektur layer CNN, *tensorflow.keras.models* untuk menyusun model sekuensial, *tensorflow.keras.callbacks* untuk mengimplementasikan *EarlyStopping* dan *ModelCheckpoint* dalam proses pelatihan, *tensorflow.keras.optimizers.Adam* untuk mengatur algoritma optimisasi saat pelatihan model.
- b. *NumPy* digunakan untuk operasi numerik seperti transformasi *array* dan manipulasi data berbasis matriks.
 - c. *Matplotlib* dan *Seaborn* digunakan untuk visualisasi grafik performa model, seperti grafik akurasi dan *loss per epoch*, serta *confusion matrix* dalam bentuk *heatmap*.
 - d. *Scikit-learn (sklearn)* digunakan untuk menghitung metrik evaluasi, seperti *confusion matrix*, *precision*, *recall*, dan *F1-score* menggunakan fungsi *confusion_matrix()* dan *classification_report()*.
2. Proses berikutnya terdapat *pre-processing* data, ditujukan untuk menyiapkan data dengan tahapan pembagian data sebesar 70% untuk data pelatihan, 30% untuk data validasi dan pengujian ,penyesuaian dimensi gambar menjadi 100x100 piksel serta menormalisasikan data dengan cara mengubah nilai piksel pada citra rentang nilai antara 0 hingga 255 kedalam rentang 0.0 hingga 1.0.



Gambar 3. 6 Contoh *Preprocessing Data*

3. Pada proses selanjutnya yaitu augmentasi data, berupa membalik citra secara horizontal dan vertikal, memutar citra secara acak hingga 30%, memperbesar citra secara acak serta penambahan noise secara ringan.



Gambar 3. 7 Contoh Augmentasi Data

4. Pada proses berikutnya dilakukan tahap perancangan model CNN dengan ketentuan:
- Lapisan awal yang menerima gambar berwarna (RGB) berukuran 100×100 piksel sebagai *input* ke dalam jaringan.
 - Penggunaan lapisan *Conv2D* serta *ReLU*.
 - Penggunaan lapisan *MaxPooling2D* guna mengurangi ukuran gambar (*downsampling*) guna mempercepat komputasi.
 - Penggunaan *GlobalAveragePooling2D* guna menghindari *overfitting*.
 - Penggunaan *Dense Layer* sebanyak 128 unit guna mengolah seluruh informasi dari fitur.
 - Penggunaan *Dropout* guna menonaktifkan sebagian *neuron* secara acak selama pelatihan untuk mencegah model menghafal data (*overfitting*) dan membuat model lebih umum.
 - Penggunaan aktivasi *Sigmoid* untuk memetakan nilai *input* ke rentang 0 dan 1 sehingga berguna untuk masalah klasifikasi biner.

```

model = models.Sequential([
    layers.Input(shape=(100, 100, 3)),
    layers.Conv2D(32, (3, 3), activation='relu'),
    layers.MaxPooling2D(),
    layers.Conv2D(64, (3, 3), activation='relu'),
    layers.MaxPooling2D(),
    layers.Conv2D(128, (3, 3), activation='relu'),
    layers.MaxPooling2D(),
    layers.GlobalAveragePooling2D(),
    layers.Dense(128, activation='relu'),
    layers.Dropout(0.5),
    layers.Dense(1, activation='sigmoid')
])

```

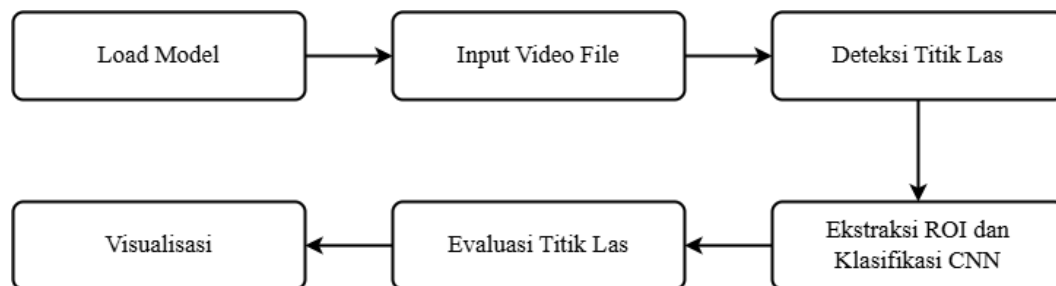
Gambar 3. 8 Potongan Program Perancangan Model CNN

5. Pada proses berikutnya dilakukan proses pelatihan model CNN menggunakan data pelatihan. Proses pelatihan dilakukan menggunakan fungsi *callbacks*, terdapat 2 *callbacks* yaitu *early_stopping* dan *checkpoint* dengan penggunaan 200 *epoch*.
6. Setelah model dibangun, dilakukan proses evaluasi melalui visualisasi-visualisasi dari hasil pelatihan model. Dalam proses ini ditujukan untuk memahami pola pelatihan model dan melihat hasil performa model dalam mengklasifikasikan citra.

Setelah tahapan pembangunan model CNN selesai, tahapan selanjutnya yaitu pengimplementasian model pada sistem deteksi *real-time* melalui *input* video. Proses deteksi dimulai dengan memanfaatkan input video guna menyediakan data visual yang akan dianalisis.

3.5 Perancangan Sistem Deteksi

Sistem deteksi yang dirancang dalam penelitian ini bertujuan untuk mengidentifikasi titik las (*weldingspot*) pada klip foto yang dijadikan video hasil inspeksi, dan menentukan status kelayakan sambungan berdasarkan jumlah serta posisi titik las yang terdeteksi. Sistem dibangun dengan pendekatan gabungan antara metode deteksi bentuk berbasis *Computer Vision* (*Hough Circle*) dan klasifikasi visual menggunakan model CNN. Tahapan utama dalam perancangan sistem ini adalah sebagai berikut.



Gambar 3. 9 Alur Sistem Deteksi

Keterangan :

1. Sistem deteksi yang dibangun dalam penelitian ini diawali dengan pemanggilan model CNN yang telah dilatih sebelumnya. Model ini dimuat dari *file* dengan format *.keras* dan digunakan untuk mengklasifikasikan citra ke dalam dua kelas utama, yaitu *weldingspot* dan *notweldspot*. Untuk memastikan kesesuaian *input*, gambar yang akan diproses harus disesuaikan ukurannya menjadi 100×100 piksel.
2. Selanjutnya, sistem menerima *input* berupa video inspeksi berformat *.mp4* yang merekam area pengelasan. Video ini diproses secara berkelanjutan, dibaca *frame* demi *frame* menggunakan pustaka *OpenCV*. Agar pemrosesan berlangsung terus-menerus, sistem secara otomatis mengulang video jika mencapai akhir.
3. Setiap *frame* dari video diolah untuk mendeteksi kandidat titik las. Proses ini dilakukan dengan mengubah gambar menjadi *grayscale* dan menerapkan *median blur* untuk mengurangi *noise*, lalu dilanjutkan dengan metode *Hough Circle Transform* guna mendeteksi bentuk lingkaran yang kemungkinan besar merupakan titik pengelasan. Lingkaran-lingkaran yang terdeteksi kemudian diekstraksi sebagai *region of interest (ROI)*, dipotong dari *frame*, diubah ukurannya, dan dinormalisasi.
4. ROI yang telah disiapkan akan diklasifikasikan menggunakan model CNN. Hasil prediksi berupa nilai probabilitas yang kemudian diterjemahkan menjadi label biner (*weldingspot*) berdasarkan ambang batas 0.5. Titik yang diklasifikasikan sebagai *weldingspot* akan dikumpulkan, dan sistem akan mengevaluasi apakah terdapat paling sedikit empat titik yang tersusun sejajar. Jika kondisi ini terpenuhi, maka *frame* dianggap memiliki sambungan las yang baik dan diberi status "OK", sedangkan jika tidak, akan diberi status "NG".

3.6 Pengujian

Pengujian sistem deteksi visual dilakukan untuk mengevaluasi kinerja model CNN dalam mengidentifikasi titik las secara otomatis dari video inspeksi. Setiap *frame* dalam video diproses untuk mendeteksi bentuk lingkaran yang berpotensi merupakan titik las menggunakan metode *Hough Circle Transform*. Lingkaran-lingkaran tersebut kemudian diekstraksi menjadi *region of interest* (ROI) dan diklasifikasikan menggunakan model CNN. Pengujian ini bertujuan untuk mengetahui seberapa baik sistem mengenali titik las yang valid serta menghindari kesalahan klasifikasi pada objek selain titik las. Setelah klasifikasi dilakukan, sistem mengevaluasi apakah titik-titik *weldingspot* yang terdeteksi memenuhi syarat kelayakan. Jika terdapat minimal empat titik yang tersusun secara horizontal dengan jarak antar titik dalam toleransi tertentu, maka sistem memberikan status “OK”. Sebaliknya, jika tidak ditemukan pola yang sesuai, maka sistem memberikan status “NG”. Hasil deteksi dan status akhir ditampilkan secara *real-time* dalam jendela visual menggunakan *OpenCV*. Pengujian dilakukan untuk memastikan bahwa semua anotasi muncul dengan jelas, termasuk lingkaran penanda titik las, label klasifikasi, dan status keseluruhan.



KARAWANG

