

LAPORAN PROYEK KECERDASAN BUATAN
Sistem Deteksi Real-Time Duku Dan Langsung

ANGGOTA KELOMPOK 8

Chindy Jopita Berutu | 231712008

Gonter Munte | 231712050



PROGRAM STUDI D-3 TEKNIK INFORMATIKA
FAKULTAS VOKASI
UNIVERSITAS SUMATERA UTARA
MEDAN
2025

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Kemajuan teknologi kecerdasan buatan, khususnya dalam bidang *computer vision*, telah membawa dampak besar terhadap berbagai sektor, termasuk pertanian. Salah satu tantangan dalam bidang ini adalah proses identifikasi dan klasifikasi buah yang masih banyak dilakukan secara manual, sehingga membutuhkan waktu dan tenaga yang tidak sedikit. Untuk mengatasi hal tersebut, diperlukan sebuah sistem yang mampu mengidentifikasi jenis buah secara otomatis dan real-time.

Proyek ini bertujuan untuk mengembangkan sistem deteksi dan klasifikasi buah, khususnya buah duku dan langsung, menggunakan teknologi deteksi objek dan klasifikasi citra berbasis deep learning. Dengan sistem ini, kamera dapat digunakan untuk mendeteksi objek yang tampak seperti buah, dan jika sesuai, dilakukan klasifikasi untuk menentukan apakah objek tersebut adalah duku, langsung, atau jenis buah lainnya.

1.2 Rumusan Masalah

Bagaimana membangun sistem yang mampu mendeteksi dan mengklasifikasikan buah duku dan langsung secara real-time dari kamera, menggunakan pendekatan berbasis YOLOv8 dan TensorFlow?

1.3 Tujuan Proyek

Proyek ini memiliki tujuan sebagai berikut:

1. Mendeteksi objek buah secara real-time dari kamera menggunakan model YOLOv8.
2. Melakukan klasifikasi terhadap objek buah tersebut apakah termasuk duku, langsung, atau bukan keduanya menggunakan model klasifikasi TensorFlow.
3. Menampilkan hasil deteksi dan klasifikasi dalam bentuk visual interaktif secara langsung.

1.4 Manfaat Proyek

Sistem yang dikembangkan dapat memberikan manfaat sebagai alat bantu dalam proses sortasi buah, riset klasifikasi produk pertanian, serta menjadi dasar untuk pengembangan sistem otomatis berbasis IoT dalam industri pertanian.

BAB 2

IMPLEMENTASI SISTEM

2.1 Arsitektur Sistem

Sistem terdiri atas dua komponen utama: model deteksi objek dan model klasifikasi citra. Proses dimulai dari pengambilan citra secara real-time melalui kamera. Setiap frame kemudian dianalisis menggunakan model YOLOv8 untuk mendeteksi objek. Jika objek yang terdeteksi termasuk dalam daftar objek buah (berdasarkan ID kelas tertentu seperti pisang, apel, jeruk, dll), maka objek tersebut dipotong dari gambar dan dikirim ke model klasifikasi untuk dianalisis lebih lanjut.

Model klasifikasi dibuat menggunakan TensorFlow dan telah dilatih secara khusus untuk mengenali dua kelas, yaitu "duku" dan "langsat". Hasil klasifikasi tersebut kemudian ditampilkan secara visual di layar dengan menggunakan OpenCV, lengkap dengan label dan tingkat kepercayaan prediksi.

2.2 Teknologi dan Perangkat Lunak

Sistem ini dikembangkan menggunakan bahasa pemrograman Python, serta pustaka-pustaka pendukung sebagai berikut:

1. OpenCV: Untuk pengambilan gambar dari kamera dan visualisasi hasil deteksi.
2. NumPy & PIL: Untuk pemrosesan gambar.
3. Menggunakan model YOLOv8n dari Ultralytics untuk deteksi objek secara cepat dan efisien. Model YOLO digunakan untuk mendeteksi bounding box serta label objek di dalam frame dari kamera secara langsung.
4. Setelah objek buah terdeteksi, bagian gambar tersebut dipotong dan diproses ulang (resize dan normalisasi). Gambar yang telah diproses diklasifikasikan menggunakan model TensorFlow (duku_langsat_model.h5) untuk membedakan antara *duku* dan *langsat*. Label klasifikasi dan tingkat kepercayaan (confidence) ditampilkan pada layar secara langsung.

```
[3] ... Found 72 images belonging to 2 classes.  
Found 17 images belonging to 2 classes.  
Class Indices: {'duku': 0, 'langsar': 1}  
Class Names: ['duku', 'langsar']
```

Gambar 2.2 model TensorFlow

5. Filr label (labels.txt): daftar nama kelas untuk klasifikasi.

2.3 Proses Deteksi dan Klasifikasi

Deteksi dimulai dengan kamera yang aktif menangkap gambar secara real-time. Setiap objek dalam frame dievaluasi apakah termasuk buah berdasarkan hasil deteksi YOLO. Bila iya, gambar objek tersebut diklasifikasikan menjadi “duku”, “langsar”, atau “tidak dikenali”. Hasil klasifikasi ditampilkan pada layar dengan kotak berwarna:

- Kuning untuk *duku*.
- Cyan untuk *langsar*.
- Oranye untuk buah umum lainnya.
- Merah untuk objek yang bukan buah.



Gambar 2.3 Proses Deteksi

BAB 3

PROJEK

3.1 Pengembangan dan Pelatihan Model AI (Klasifikasi Duku/Langsar)

A. Pembangunan Arsitektur Model CNN

Model AI utama dirancang menggunakan arsitektur Convolutional Neural Network (CNN) dengan Keras Sequential API. Arsitektur model ini terdiri dari:

- Lapisan konvolusi (`Conv2D`) untuk mengekstraksi fitur visual dari gambar,
- Lapisan pooling (`MaxPooling2D`) untuk mereduksi dimensi dan menjaga fitur penting,
- Lapisan flatten untuk mengubah hasil dari dua dimensi menjadi satu vektor,
- Beberapa lapisan dense dan dropout untuk klasifikasi akhir serta pencegahan overfitting.

B. Kompilasi Model

Sebelum model dilatih, dilakukan kompilasi dengan menentukan:

- Optimizer: digunakan Adam, untuk menyesuaikan bobot selama pelatihan,
- Loss function: digunakan Categorical Crossentropy, yang cocok untuk klasifikasi dua kelas,
- Metrik: akurasi digunakan sebagai metrik utama untuk mengevaluasi performa model selama pelatihan.

C. Proses Pelatihan Model

Pelatihan dilakukan secara bertahap (beberapa epoch) di Jupyter Notebook. Dataset dibagi menjadi data pelatihan dan validasi. Selama pelatihan, grafik akurasi dan loss dimonitor untuk memastikan model belajar dengan baik dan tidak overfitting. Jika diperlukan, early stopping atau penyesuaian arsitektur dapat diterapkan.

D. Penyimpanan Model Terlatih

Setelah pelatihan selesai dan model menunjukkan performa yang baik, seluruh struktur model beserta bobot hasil pelatihan disimpan dalam file `.h5` (misalnya `duku_langsat_model.h5`). Label kelas juga disimpan ke file `labels.txt`. Kedua file ini akan digunakan untuk keperluan klasifikasi dan integrasi aplikasi.

3.2 Implementasi AI dan Pengembangan Aplikasi (Python Murni)

A. Pengembangan Skrip Klasifikasi Mandiri (`classify_image.py`)

Skrip ini digunakan untuk menguji kemampuan model secara offline. Ia memuat model `.h5`, membaca gambar dari file lokal, melakukan pra-pemrosesan, lalu menghasilkan prediksi klasifikasi sebagai "duku" atau "langsar" disertai persentase keyakinannya. Hasilnya ditampilkan langsung di terminal.

B. Pengembangan Skrip Deteksi Real-time (`realtime_detector.py`)

Aplikasi Python desktop ini menggunakan OpenCV untuk mengakses kamera secara langsung dan mengintegrasikan deteksi objek real-time. YOLOv8 digunakan untuk mendeteksi objek secara umum (misalnya buah), lalu bagian objek tersebut dipotong dan dikirim ke model klasifikasi duku/langsat untuk identifikasi lebih lanjut.

C. Integrasi Model AI ke Aplikasi

Pada kedua skrip (klasifikasi gambar dan deteksi real-time), model yang telah dilatih (`duku_langsat_model.h5`) dan file label dimuat ke dalam memori saat aplikasi dijalankan. Hal ini memungkinkan proses klasifikasi langsung tanpa perlu melatih ulang model setiap kali.

D. Implementasi Deteksi dan Visualisasi

Dalam skrip real-time, kotak pembatas (bounding box) digambar di sekitar objek yang terdeteksi. Setiap objek diberi label hasil dari model YOLO (misalnya “buah”) dan hasil klasifikasi akhir (“duku” atau “langsat”). Warna dan teks label disesuaikan agar memudahkan pengguna membedakan hasil klasifikasi secara visual. Tampilan kamera disesuaikan dalam jendela tertentu agar optimal untuk pengguna.

3.3 Pengujian dan Evaluasi Berkesinambungan

A. Pengujian Fungsional

Selama proses pengembangan, dilakukan pengujian berulang menggunakan:

- Skrip `classify_image.py` untuk menguji performa klasifikasi gambar statis,
- Skrip `realtime_detector.py` untuk pengujian langsung melalui kamera.

Hasil pengujian digunakan untuk mengevaluasi apakah model memberikan prediksi yang akurat dan aplikasi berjalan sesuai ekspektasi.

B. Evaluasi Performa

Setiap hasil klasifikasi dan deteksi diamati untuk mengukur akurasi, waktu respons, dan konsistensi. Jika ditemukan kelemahan—seperti prediksi salah atau gagal mendeteksi objek—maka dilakukan analisis untuk memperbaiki dataset atau parameter model. Evaluasi ini dilakukan secara berulang sebagai bagian dari iterasi pengembangan yang berkesinambungan.

BAB 4

PENUTUP

3.1 Kesimpulan

Proyek ini berhasil mengimplementasikan sistem deteksi dan klasifikasi buah duku dan langsung secara real-time menggunakan kombinasi model YOLOv8 untuk deteksi objek dan model TensorFlow untuk klasifikasi citra. Sistem ini mampu mengidentifikasi objek buah dari kamera dan menentukan apakah objek tersebut merupakan duku, langsung, atau buah umum lainnya, serta memberikan umpan balik visual secara langsung dengan tingkat kejelasan yang baik.

Keunggulan utama dari sistem ini adalah kemampuannya untuk bekerja secara real-time dan hanya memproses objek-objek yang relevan, yaitu buah-buahan, sehingga efisien dan tidak membebani proses klasifikasi secara keseluruhan. Visualisasi yang disediakan juga cukup informatif, memberikan label dan warna yang berbeda berdasarkan hasil klasifikasi, yang memudahkan dalam interpretasi hasil.