Jurnal Elektronika dan Teknik Informatika Terapan (JENTIK)
Volume 4 Nomor 2 Juni 2026
e-ISSN: 2988-0874; p-ISSN: 2988-0866, Hal 176-198
DOI: https://doi.org/10.59061/jentik.v4i2.1564
Tersedia: https://e-journal.poltek-kampar.ac.id/index.php/JENTIK

Fish Check: Sistem Berbasis *Convolutional Neural Network* untuk Menilai Kelayakan Konsumsi Ikan Laut di Desa Weru, Kecamatan Paciran, Kabupaten Lamongan

Santi Febrianti[1*], Aisyah Nur Nabila[2], Ratih Berliana[3], Nuke Amalia[4], Rudy Suryadi[5], Riyan Bagus Prihandanu[6]

[1,2] Teknik Bangunan Kapal, Politeknik Perkapalan Negeri Surabaya, Indonesia
[3,4,5,6] Teknik Permesinan Kapal, Politeknik Perkapalan Negeri Surabaya, Indonesia
Email: santifebrianti@ppns.ac.id[1*]
Korespondensi penulis: santifebrianti@ppns.ac.id

Abstract. *Conventional assessment of fish freshness remains subjective, resulting in inconsistent evaluations. Advances in computer vision and Convolutional Neural Networks (CNNs) offer an opportunity to develop automated and objective approaches for assessing fish freshness through mobile devices. This study developed Fish Check, a smartphone-based application that utilizes the MobileNetV2 architecture to classify the freshness status of marine fish from digital images captured using a smartphone camera. A dataset comprising 2,400 marine fish images was collected. Image acquisition was conducted under varying lighting conditions, camera angles, and object positions to represent practical field conditions. The proposed approach involved image preprocessing, data augmentation, transfer learning-based model training, and performance evaluation using Accuracy, Precision, Recall, F1-score, and a Confusion Matrix. The optimized MobileNetV2 model was subsequently integrated into a React Native mobile application using TensorFlow Lite to enable efficient real-time inference directly on smartphones. The experimental results showed that the proposed model achieved an Accuracy of 97.22%, Precision of 97.50%, Recall of 97.22%, and F1-score of 97.35% demonstrate that the model can effectively distinguish between fresh and non-fresh marine fish. Overall, Fish Check provides a practical approach for real-time fish freshness assessment by combining a locally collected marine fish image dataset with an efficient deep learning architecture optimized for mobile deployment*.

Keywords: *Computer Vision, Convolutional Neural Network, Fish Freshness, MobileNetV2, TensorFlow Lite*

Abstrak. Penilaian kesegaran ikan secara konvensional masih bersifat subjektif sehingga menghasilkan evaluasi yang tidak konsisten. Kemajuan dalam bidang *computer vision* dan *Convolutional Neural Networks* (CNN) membuka peluang untuk mengembangkan pendekatan otomatis dan objektif dalam menilai kesegaran ikan melalui perangkat seluler. Penelitian ini mengembangkan *Fish Check*, sebuah aplikasi berbasis ponsel pintar yang memanfaatkan arsitektur MobileNetV2 untuk mengklasifikasikan tingkat kesegaran ikan laut berdasarkan citra digital yang diambil menggunakan kamera ponsel. Dataset yang terdiri dari 2.400 citra ikan laut. Pengambilan citra dilakukan dengan variasi kondisi pencahayaan, sudut kamera, dan posisi objek untuk merepresentasikan kondisi lapangan yang sesungguhnya. Pendekatan yang diusulkan mencakup pra-pemrosesan citra, augmentasi data, pelatihan model berbasis *transfer learning*, serta evaluasi kinerja menggunakan metrik Akurasi, Presisi, *Recall*, *F1-score*, dan *Confusion Matrix*. Model MobileNetV2 yang telah dioptimalkan kemudian diintegrasikan ke dalam aplikasi seluler React Native menggunakan TensorFlow Lite untuk memungkinkan inferensi *real-time* yang efisien langsung pada ponsel pintar. Hasil eksperimen menunjukkan bahwa model yang diusulkan mencapai akurasi sebesar 97,22%, presisi 97,50%, recall 97,22%, dan skor F1 sebesar 97,35% yang membuktikan bahwa model mampu membedakan ikan laut segar dan tidak segar secara efektif. Secara keseluruhan, *Fish Check* menawarkan pendekatan yang praktis untuk penilaian kesegaran ikan secara *real-time* dengan memadukan dataset citra ikan laut yang dikumpulkan secara lokal serta arsitektur *deep learning* efisien yang dioptimalkan untuk penggunaan pada perangkat seluler.

Kata kunci: *Computer Vision, Convolutional Neural Network,* Kesegaran Ikan, *MobileNetV2, TensorFlow Lite*

Naskah Masuk: 01 Mei 2026; Revisi: 15 Mei 2026; Diterima: 25 Juni 2026; Terbit: 30 Juni 2026

1. PENDAHULUAN

Latar Belakang

Kabupaten Lamongan merupakan salah satu pusat perikanan tangkap laut di Jawa Timur, dengan masyarakat pesisir yang bergantung pada hasil perikanan. Desa Weru, Kecamatan Paciran, memiliki sumber daya ikan yang melimpah dan berperan dalam pemenuhan protein, gizi, serta perekonomian masyarakat (Hanifa et al., 2023). Namun, ikan mudah mengalami penurunan mutu akibat penanganan dan penyimpanan yang kurang baik. Pengendalian suhu yang tidak optimal dapat mempercepat pertumbuhan mikroba dan pembentukan histamin yang berisiko terhadap keamanan pangan (Gazpersz et al., 2024).

Kesegaran ikan umumnya dinilai secara organoleptik melalui kondisi mata, insang, kulit, dan daging. Meskipun sederhana dan murah, metode ini bergantung pada pengalaman dan subjektivitas penilai sehingga hasilnya dapat tidak konsisten (Anas et al., 2021). Perkembangan kecerdasan buatan, khususnya *computer vision* berbasis *Convolutional Neural Network* (CNN), menawarkan alternatif penilaian otomatis dengan memanfaatkan fitur visual seperti warna, tekstur, dan morfologi mata (Mahendra & Faurina, 2024). Arsitektur VGG16, *ResNet*, *EfficientNet*, dan *MobileNetV2* telah menunjukkan kinerja yang menjanjikan dalam klasifikasi ikan, dengan *MobileNetV2* memiliki keunggulan berupa ukuran model dan kebutuhan komputasi yang rendah (Al-Ghiffary et al., 2023; Gunawan et al., 2023; Suhendra et al., 2025).

Namun, penelitian sebelumnya masih banyak menggunakan dataset laboratorium yang kurang merepresentasikan kondisi lapangan, sementara dataset primer ikan laut dari pesisir Indonesia masih terbatas. Penerapan CNN secara *offline* pada perangkat seluler juga masih jarang karena sebagian sistem bergantung pada *server* eksternal (Hanifa et al., 2023). Kondisi tersebut menunjukkan perlunya sistem yang ringan, adaptif terhadap kondisi lapangan, dan mampu melakukan inferensi secara *on device*.

Berdasarkan permasalahan tersebut, penelitian ini mengembangkan *Fish Check*, aplikasi ponsel untuk klasifikasi kesegaran ikan laut secara *real time* berbasis *edge computing*. Sistem menggunakan *MobileNetV2* dengan *transfer learning* yang dilatih menggunakan dataset primer dari Desa Weru dengan variasi kondisi pengambilan citra di lapangan. Model dioptimalkan menggunakan *TensorFlow Lite* dan diintegrasikan ke *React Native* sehingga dapat berjalan secara *on device* tanpa koneksi internet maupun

server (Sandler et al., 2019). Penelitian ini berkontribusi melalui pengembangan model ringan, penyediaan dataset lokal berbasis kondisi lapangan, serta implementasi klasifikasi *real time* yang praktis dan *non-destructive* untuk mendukung pengendalian mutu ikan (Jaydish & Chelladurai, 2026).

2. KAJIAN TEORITIS

Computer Vision

Computer vision merupakan cabang kecerdasan buatan yang memungkinkan komputer memperoleh dan menganalisis informasi visual dari citra atau video secara otomatis (Szeliski, 2022). Teknologi ini mendukung otomatisasi inspeksi visual melalui tahapan pra-pemrosesan, ekstraksi fitur, dan klasifikasi untuk mengenali pola serta tekstur secara objektif (Sonka, Hlavac & Boyle, 2014; Gonzalez & Woods, 2018). Penerapannya pada berbagai sektor dapat meningkatkan efisiensi dan konsistensi evaluasi kualitas (Prince, 2012), sehingga sesuai untuk penilaian objek secara presisi dan *non-destructive* (Forsyth & Ponce, 2012).

Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) merupakan arsitektur *deep learning* yang dirancang untuk memproses data berbentuk kisi dua dimensi, seperti citra digital (LeCun et al., 2015). CNN mampu mengekstraksi fitur secara otomatis dan hierarkis melalui lapisan konvolusi tanpa rekayasa fitur manual (Goodfellow et al., 2016). Penggunaan *weight sharing* dan *pooling* mengurangi jumlah parameter sekaligus memberikan ketahanan terhadap perubahan posisi objek (Chollet, 2021). Kemampuan tersebut menjadikan CNN banyak digunakan dalam klasifikasi citra modern (Zhang et al., 2021).

Arsitektur MobileNetV2

MobileNetV2 merupakan arsitektur CNN yang dirancang untuk perangkat dengan sumber daya terbatas, seperti perangkat seluler (Sandler et al., 2018). Arsitektur ini menggunakan *inverted residual block*, *linear bottleneck*, dan *depthwise separable convolution* untuk mengurangi parameter serta beban komputasi dengan tetap mempertahankan kinerja model (Elfwing et al., 2021; Al-Quraishi et al., 2022). Efisiensi tersebut menjadikan *MobileNetV2* sesuai sebagai *backbone* untuk klasifikasi *real-time* pada perangkat *edge* (Shorten & Khoshgoftaar, 2021; Sinha & Mukherjee, 2023).

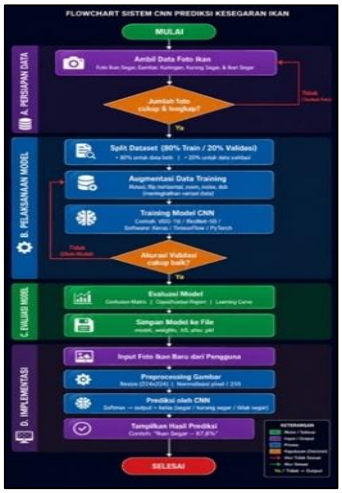
TensorFlow Lite

TensorFlow Lite (TFLite) merupakan kerangka kerja *edge computing* yang memungkinkan model pembelajaran mesin dijalankan langsung pada perangkat seluler dan perangkat tertanam (David et al., 2021). *TFLite* mendukung optimasi seperti kuantisasi dan *pruning* untuk mengurangi ukuran model dan beban komputasi (Gupta et al., 2022). Inferensi secara lokal juga dapat mengurangi latensi dan ketergantungan pada *cloud* (Jain et al., 2023). Integrasinya dengan *React Native* memungkinkan penerapan model AI secara efisien pada aplikasi seluler (Ray, 2022; Perez et al., 2024).

3. METODE PENELITIAN

Desain Penelitian

Penelitian ini menggunakan pendekatan eksperimental untuk mengembangkan dan mengevaluasi sistem klasifikasi kesegaran ikan laut berbasis CNN dengan arsitektur *MobileNetV2*. Tahapan penelitian meliputi akuisisi dataset, pengembangan model, implementasi aplikasi seluler, dan evaluasi sistem. Citra ikan diambil menggunakan ponsel pintar dengan variasi pencahayaan, sudut, dan latar belakang, kemudian dilakukan pra-pemrosesan, augmentasi, dan transfer learning menggunakan *TensorFlow* (Chollet, 2021). Model terbaik dikonversi ke *TensorFlow Lite* (TFLite) dan diintegrasikan ke *React Native* untuk *on device inference*. Kinerja model dan aplikasi dievaluasi berdasarkan akurasi, presisi, *recall, F1-score*, *confusion matrix*, waktu inferensi, dan kemampuan klasifikasi *real time.*



Gambar 1 Alur kerja keseluruhan sistem klasifikasi kesegaran ikan laut berbasis CNN yang diusulkan menggunakan *MobileNetV2*

Akuisisi Dataset

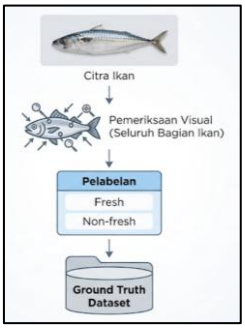
Penelitian ini menggunakan 372 citra primer ikan laut yang dikumpulkan di Desa Weru, Kecamatan Paciran, Kabupaten Lamongan, terdiri atas Ikan Kembung (*Rastrelliger brachysoma*), Ikan Kurisi (*Nemipterus bathybius*), dan Ikan Gerabah (*Johnius heterolepis*). Citra diklasifikasikan menjadi segar dan tidak segar dengan variasi pencahayaan, sudut, jarak, dan latar belakang. Pengambilan citra difokuskan pada mata ikan karena merupakan salah satu indikator visual kesegaran (Faurina & Mahendra, 2024). Selanjutnya, citra disaring dan diproses melalui prapemrosesan serta augmentasi. Model dirancang untuk melakukan klasifikasi biner dengan kelas Segar dan Tidak Segar. Ketiga spesies digunakan untuk memperkaya variasi fitur visual, seperti warna, bentuk mata, dan tekstur sisik, sehingga model dapat mengenali karakteristik kesegaran secara lebih umum dan mengurangi *overfitting*. Output model menggunakan satu neuron dengan fungsi aktivasi Sigmoid untuk menghasilkan probabilitas kelayakan konsumsi. Pembagian dataset dilakukan menggunakan teknik pengambilan sampel acak *berstrata (stratified random sampling)* dengan proporsi 70:15:15 dari total 2.400 citra akhir. Jumlah dataset disajikan secara eksplisit pada masing-masing subset sebagai berikut:

- Data Pelatihan (*Training Set*, 70%): Berjumlah 1.680 citra, yang terdiri atas 840 citra kategori segar dan 840 citra kategori tidak segar. *Subset* ini digunakan untuk penyesuaian bobot (weight optimization) pada arsitektur *MobileNetV2*.
- Data Validasi (*Validation Set*, 15%): Berjumlah 360 citra, yang terdiri atas 180 citra kategori segar dan 180 citra kategori tidak segar. *Subset* ini berfungsi untuk pemantauan *hyperparameter tuning* dan pencegahan overfitting selama proses pelatihan.
- Data Pengujian (*Test Set*, 15%): Berjumlah 360 citra, yang terdiri atas 180 citra kategori segar dan 180 citra kategori tidak segar. Subset ini diisolasi sepenuhnya dari proses pelatihan untuk mengukur performa evaluasi akhir model (*confusion matrix*, akurasi, presisi, recall, dan F1-score).

Anotasi Data

Klasifikasi utama pada penelitian ini berfokus pada klasifikasi biner tingkat kesegaran, yaitu kelas Segar dan Tidak Segar. Tiga jenis ikan laut yang digunakan dalam dataset (Ikan Kembung, Ikan Kurisi, dan Ikan Gerabah) tidak ditujukan untuk pengenalan/spesifikasi spesies oleh model, melainkan dimanfaatkan sebagai variasi

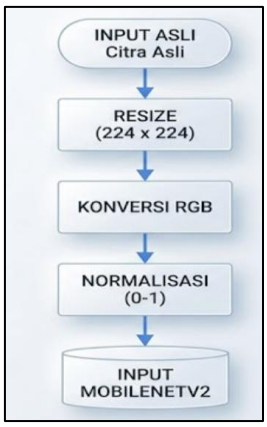
sampel untuk memperkaya fitur visual (seperti tekstur mata, kulit, dan warna). Variasi tiga jenis ikan ini bertujuan agar model *MobileNetV2* memiliki kemampuan generalisasi yang kuat dan tidak mengalami *overfitting* pada satu spesies tertentu. Lebih lanjut, penentuan status "Layak Konsumsi" atau "Tidak Layak Konsumsi" dalam aplikasi dipetakan secara langsung dari hasil prediksi kesegaran visual berbasis ciri organoleptik mata ikan (SNI 2729:2013). Prediksi ini berfungsi sebagai sistem pendukung keputusan awal berbasis indikator fisik visual, dan bukan merupakan analisis laboratorium penuh terhadap pembentukan histamin atau cemaran mikrobiologis.



Gambar 2 Alur kerja proses anotasi citra ikan untuk menghasilkan dataset *ground truth*

Prapemrosesan Citra

Sebelum pelatihan model, semua citra diubah ukurannya menjadi 224 × 224 piksel, dikonversi ke format RGB, dan dinormalisasi ke rentang 0–1. Pengaturan ini diterapkan untuk memenuhi persyaratan *input MobileNetV2* serta mendukung pelatihan model yang stabil (Sandler et al., 2018). Citra yang telah diproses awal tersebut kemudian digunakan untuk augmentasi data dan pengembangan model.



Gambar 3 Alur kerja pra-pemrosesan citra, mencakup pengubahan ukuran, konversi RGB, normalisasi, dan penyiapan input untuk *MobileNetV2*.

Augmentasi Data

Pengumpulan data awal (*raw data*) menghasilkan 372 citra asli yang dikumpulkan secara langsung dari perikanan pesisir Desa Weru, Kabupaten Lamongan, terdiri dari 186 citra ikan segar dan 186 citra ikan tidak segar. Untuk mengatasi keterbatasan jumlah sampel asli serta memitigasi risiko *overfitting* selama proses pelatihan model, dilakukan eksplorasi dan augmentasi data sintetis pada *subset* data pelatihan. Augmentasi diterapkan dengan menerapkan kombinasi transformasi spasial dan pencahayaan, meliputi rotasi acak (hingga 30°), pembalikan horizontal *(horizontal flip)*, pergeseran lebar dan tinggi *(width/height shift* hingga 20%), serta penyesuaian kecerahan *(brightness adjustment).* Melalui proses replikasi dan variasi geometris tersebut, setiap sampel citra asli diekspansi sehingga menghasilkan dataset akhir yang seimbang *(balanced dataset)* sebanyak 2.400 citra, yang terdiri atas 1.200 citra kategori segar dan 1.200 citra kategori tidak segar.

Arsitektur *MobileNetV2*

MobileNetV2 dipilih sebagai *backbone* CNN karena menawarkan keseimbangan antara efisiensi komputasi dan kinerja klasifikasi. Desainnya yang ringan membutuhkan sumber daya komputasi yang lebih sedikit dibandingkan banyak arsitektur CNN konvensional, sehingga cocok untuk penerapan pada perangkat seluler (Sandler et al., 2018). Arsitektur ini mengintegrasikan *depthwise separable convolution*, *inverted residual block*, dan *linear bottleneck* untuk mengurangi kompleksitas komputasi sembari tetap mempertahankan ekstraksi fitur yang efektif. Dalam penelitian ini, model menerima input berupa citra RGB berukuran 224 × 224 piksel, dan bagian *classification head* aslinya diganti dengan lapisan keluaran biner yang merepresentasikan kategori ikan segar dan ikan tidak segar.

Strategi *Transfer Learning*

Transfer learning diterapkan menggunakan *MobileNetV2* yang telah dilatih sebelumnya pada dataset *ImageNet.* Lapisan konvolusi hasil pelatihan awal tersebut dipertahankan sebagai ekstraktor fitur, sementara bagian klasifikasi (*classification head*) aslinya diganti dengan lapisan keluaran biner khusus (Pan & Yang, 2010; Sandler et al., 2018). Model yang telah dimodifikasi tersebut kemudian disesuaikan (*fine-tuned*) menggunakan dataset citra ikan. Kinerja pada tahap validasi digunakan untuk memilih model dengan performa terbaik. Pendekatan ini mengurangi kebutuhan pelatihan serta

memungkinkan model untuk mempelajari fitur visual yang relevan dari dataset yang relatif terbatas.

Konfigurasi *Training*

Dataset akhir berjumlah 2.400 citra dibagi secara *stratified random sampling* menjadi subset pelatihan (70% atau 1.680 citra), validasi (15% atau 360 citra), dan pengujian (15% atau 360 citra). Pelatihan arsitektur *MobileNetV2* dilakukan menggunakan kerangka kerja *TensorFlow* dengan pendekatan dua tahap *(two-stage transfer learning)*. Tahap pertama *(feature extraction)* dilakukan dengan membekukan seluruh *base-layer MobileNetV2* (154 layer di-*freeze*) dan melatih lapisan klasifikasi baru selama 23 *epoch* dengan *learning rate* $1x10^{-3}$. Tahap kedua *(fine-tuning)* dilakukan dengan membuka pembekuan beberapa layer teratas dan melatih model selama 10 *epoch* tambahan dengan *learning rate* yang lebih kecil, yaitu $1x10^{-5}$. Kedua tahap pelatihan menggunakan ukuran *batch (batch size)* sebesar 32, pengoptimasi *(optimizer)* Adam, dan fungsi kerugian *Binary Cross Entropy*. Akurasi serta nilai kerugian *(loss)* pada tahap pelatihan dan validasi dipantau secara berkelanjutan, di mana model dengan kinerja validasi terbaik dipilih untuk evaluasi akhir. Augmentasi data yang mencakup rotasi acak, *zooming*, pergeseran spasial, dan pembalikan horizontal diterapkan untuk memperluas variasi data. Langkah-langkah pra-pemrosesan ini sangat penting untuk meningkatkan kemampuan generalisasi model dan memitigasi potensi *overfitting* selama fase pelatihan. Selain itu, penggunaan teknik *deep learning* yang canggih ini memungkinkan identifikasi indikator visual yang krusial, seperti karakteristik mata dan insang, yang merupakan penanda utama dalam mengevaluasi kualitas makanan laut (Akgül et al., 2023)

Evaluasi Model

Model akhir dievaluasi menggunakan set pengujian independen yang tidak digunakan selama tahap pelatihan atau validasi. Kinerja diukur menggunakan Akurasi, Presisi, *Recall*, skor-F1, dan *Confusion Matrix* untuk menilai akurasi klasifikasi serta kemampuan generalisasi terhadap citra yang belum pernah dilihat sebelumnya. Akurasi merepresentasikan proporsi sampel yang diklasifikasikan dengan benar. Presisi menunjukkan keandalan prediksi positif, sedangkan *Recall* mengukur kemampuan model dalam mengidentifikasi sampel positif. Skor-F1 merepresentasikan keseimbangan antara Presisi dan *Recall*. *Confusion Matrix* menyajikan distribusi prediksi *True Positive* (TP),

True Negative (TN), *False Positive* (FP), dan *False Negative* (FN) (Setiadi et al., 2022; Yildiz et al., 2024). Rumus matematis untuk metrik evaluasi tersebut sebagai berikut:

- *True Positive* (TP): Sampel positif yang diklasifikasikan dengan benar.
- *True Negative* (TN): Sampel negatif yang diklasifikasikan dengan benar.
- *False Positive* (FP): Sampel negatif yang secara keliru diklasifikasikan sebagai positif.
- *False Negative* (FN): Sampel positif yang secara keliru diklasifikasikan sebagai negatif.

Konversi *TensorFlow Lite*

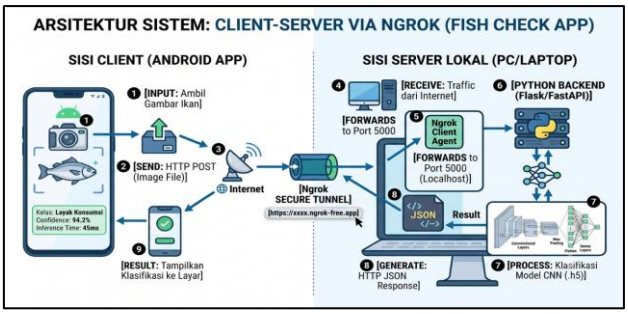
Model dengan performa validasi terbaik dikonversi ke format *TensorFlow Lite* (.tflite) menggunakan *TensorFlow Lite Converter*. Konversi ini mengurangi ukuran model, kebutuhan memori, dan beban komputasi, sekaligus mempertahankan model yang telah dilatih untuk keperluan inferensi pada perangkat seluler. Model .tflite yang dihasilkan kemudian digunakan sebagai mesin inferensi untuk aplikasi Fish Check.

Deployment React Native

Model *TensorFlow Lite* diintegrasikan ke dalam aplikasi Android Fish Check menggunakan *React Native*. Aplikasi ini melakukan inferensi secara langsung pada ponsel pintar tanpa memerlukan pemrosesan berbasis *cloud*. Saat beroperasi, pengguna mengambil gambar ikan menggunakan kamera ponsel pintar. Gambar tersebut secara otomatis diproses awal dan diteruskan ke model *TensorFlow Lite*. Selanjutnya, aplikasi menampilkan prediksi kategori kesegaran, baik segar maupun tidak segar, beserta skor tingkat keyakinannya.

4. HASIL DAN PEMBAHASAN

Arsitektur Sistem

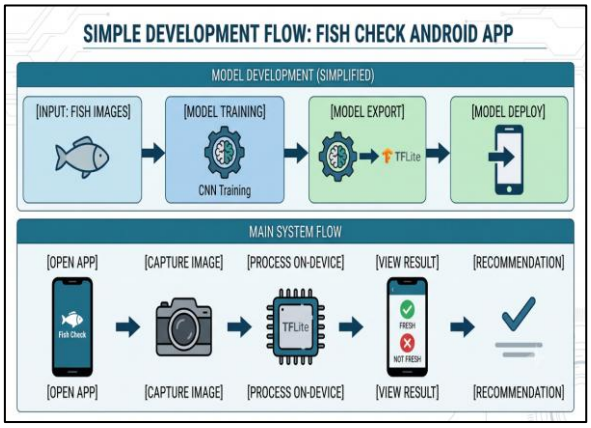


Gambar 4 Arsitektur Sistem

Gambar 4 menunjukkan arsitektur sistem aplikasi Fish Check yang terdiri atas tahapan pengembangan model dan alur kerja sistem. Pengembangan model diawali dengan pengumpulan dan pra-pemrosesan dataset citra, dilanjutkan dengan pelatihan dan evaluasi model CNN untuk memperoleh model dengan kinerja terbaik. Dataset yang digunakan mencakup tiga jenis ikan (Kembung, Kurisi, dan Gerabah) sebagai variasi bentuk visual untuk memperkuat generalisasi model. Meskipun demikian, model tidak melakukan klasifikasi jenis ikan, melainkan melakukan klasifikasi biner tingkat kesegaran, yaitu kategori segar dan tidak segar.

Model terpilih kemudian dikonversi ke format *TensorFlow Lite* (TFLite) agar dapat diintegrasikan dan dijalankan secara efisien pada perangkat Android. Pada alur sistem utama, pengguna mengambil citra ikan menggunakan kamera ponsel pintar. Citra tersebut melalui tahap pra-pemrosesan sebelum diproses oleh model CNN yang terintegrasi melalui TFLite. Model kemudian mengklasifikasikan citra ke dalam kategori segar atau tidak segar. Hasil klasifikasi ditampilkan pada antarmuka aplikasi disertai nilai tingkat kepercayaan *(confidence score)* sebagai informasi pendukung bagi pengguna. Selanjutnya, aplikasi memberikan rekomendasi berdasarkan hasil klasifikasi untuk mendukung pengambilan keputusan terkait konsumsi, distribusi, dan penanganan ikan. Dalam hal ini, status segar dipetakan menjadi keputusan rekomendasi "layak konsumsi" berdasarkan indikator fisik visual mata ikan. *Output* ini dirancang sebagai penilai awal *(first-line screening)* secara non-destruktif di lapangan dan tidak menggantikan pengujian laboratorium mikrobiologis.

Proses inferensi dilakukan secara lokal *(on-device)* pada ponsel pintar sehingga tidak bergantung pada koneksi internet maupun layanan komputasi awan eksternal. Pendekatan ini memungkinkan citra diproses tanpa dikirimkan ke *server* serta mendukung pemrosesan dengan latensi yang lebih rendah. Penggunaan TFLite juga memungkinkan model disesuaikan dengan keterbatasan sumber daya komputasi dan memori perangkat seluler. Dengan demikian, arsitektur Fish Check dirancang sebagai sistem klasifikasi yang ringan dan praktis untuk mendukung penilaian kesegaran ikan pada kondisi penggunaan di lapangan.



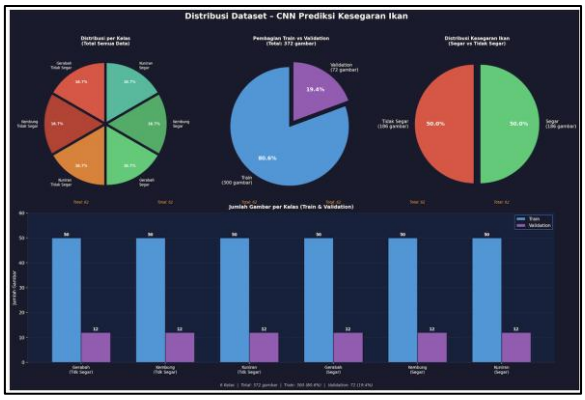
Gambar 5 *System Flow*

Gambaran Umum Dataset

Penelitian ini menggunakan dataset citra primer yang diperoleh dari ikan laut yang dikumpulkan di Desa Weru, Kecamatan Paciran, Kabupaten Lamongan, Indonesia (Gambar 6). Pengambilan citra dilakukan menggunakan kamera ponsel pintar, dengan mata ikan sebagai area visual utama yang menjadi fokus pengamatan. Perubahan pada tampilan mata memberikan informasi visual penting untuk menentukan tingkat kesegaran ikan. Setelah pengambilan citra, setiap sampel diberi label secara manual ke dalam dua kelas, yaitu segar (mewakili ikan yang dianggap layak konsumsi) dan tidak segar (mewakili ikan yang dianggap tidak layak konsumsi).

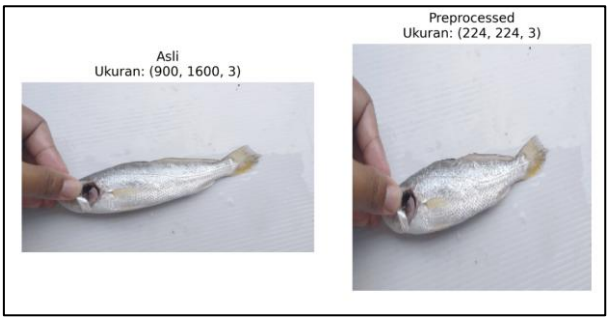
Pengumpulan data awal *(raw data)* menghasilkan 372 citra asli (186 segar dan 186 tidak segar) yang dikumpulkan secara langsung dari perikanan pesisir Desa Weru, Kabupaten Lamongan. Untuk memitigasi risiko *overfitting* serta memperluas variasi data, dilakukan augmentasi sintetis melalui rotasi acak, pembalikan horizontal *(horizontal flip)*, pergeseran spasial, dan penyesuaian kecerahan hingga menghasilkan total 2.400 citra seimbang (1.200 citra segar dan 1.200 citra tidak segar). Dataset ini kemudian dibagi menggunakan teknik pengambilan sampel acak berstrata *(stratified random sampling)* dengan rasio 70:15:15. Pembagian tersebut menghasilkan rincian aktual subset yang meliputi: data pelatihan *(training set)* sebanyak 1.680 citra (840 segar dan 840 tidak segar), data validasi *(validation set)* sebanyak 360 citra (180 segar dan 180 tidak segar), serta data pengujian *(test set)* sebanyak 360 citra (180 segar dan 180 tidak segar). Pembagian berstrata ini menjaga proporsi sampel yang seimbang di setiap subset untuk mencegah bias kelas dan memberikan dasar evaluasi kinerja yang konsisten.

Proses pengambilan gambar dirancang untuk merepresentasikan kondisi riil yang ditemui di lapangan saat aplikasi digunakan. Dataset mencakup variasi intensitas pencahayaan alami, sudut pengambilan gambar, jarak objek, dan latar belakang yang tidak seragam dari 3 spesies ikan laut (Ikan Kembung, Ikan Kurisi, dan Ikan Gerabah). Keragaman visual ini memaparkan model *Convolutional Neural Network* (CNN) pada berbagai karakteristik fitur selama proses pembelajaran guna meningkatkan kemampuan generalisasinya. Dengan demikian, model *MobileNetV2* yang dioptimalkan menggunakan *TensorFlow Lite* lebih siap untuk melakukan klasifikasi biner tingkat kesegaran secara instan dan *offline* saat diimplementasikan dalam aplikasi seluler berbasis Android di lingkungan perikanan pesisir.



Gambar 6 Gambaran Umum Dataset

Prapemrosesan Citra

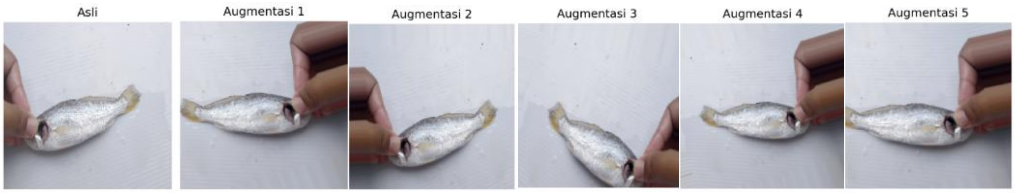


Gambar 7 Alur Kerja Prapemrosesan Citra

Sebelum digunakan dalam proses pelatihan model *MobileNetV2*, citra terlebih dahulu melalui tahap pra-pemrosesan untuk menyeragamkan karakteristik data masukan. Tahap ini dilakukan karena citra yang diperoleh memiliki variasi resolusi, pencahayaan, dan kondisi visual. Setiap citra diubah ukurannya menjadi 224 × 224 piksel, dikonversi ke ruang warna RGB, kemudian dinormalisasi dengan mengubah nilai intensitas piksel

ke dalam rentang 0–1. Proses tersebut menghasilkan citra dengan dimensi, format, dan skala nilai piksel yang konsisten sesuai dengan kebutuhan masukan *MobileNetV2*. Standardisasi ini membantu mengurangi pengaruh variasi perangkat kamera dan kondisi lingkungan saat pengambilan citra, sehingga model dapat lebih fokus dalam mengekstraksi karakteristik visual yang relevan. Citra yang telah melalui prapemrosesan selanjutnya digunakan pada tahap augmentasi dan pelatihan CNN untuk mendukung proses klasifikasi serta meningkatkan kemampuan generalisasi model terhadap citra ikan pada kondisi penggunaan aplikasi Android.

Visualisasi Data Augmentasi



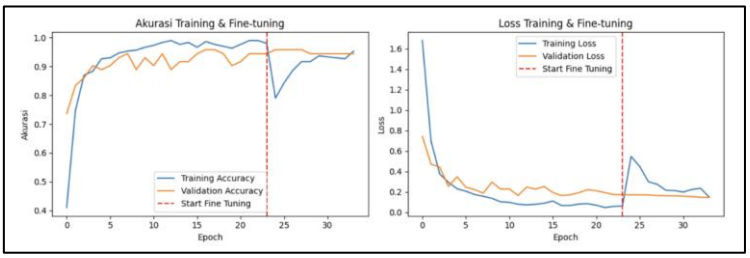
Gambar 8 Augmentasi Data

Untuk meningkatkan ketahanan model terhadap variasi dunia nyata dalam pengambilan gambar menggunakan ponsel pintar konsumen seperti pencahayaan, jarak, dan sudut pengambilan, strategi augmentasi data diterapkan secara eksklusif pada set pelatihan, sementara data validasi dan pengujian dibiarkan tanpa manipulasi (Gambar 8). Berbagai transformasi termasuk rotasi acak, pergeseran spasial, penskalaan, pembalikan horizontal, dan penyesuaian kecerahan diterapkan dengan tetap mempertahankan label keasgararan (*freshness*) sebagai *ground-truth* serta ciri-ciri morfologis utamanya. Berfungsi sebagai regularisator yang efektif untuk mencegah *overfitting*, alur kerja augmentasi ini memungkinkan jaringan untuk mempelajari representasi visual yang invarian, sehingga menjamin kemampuan generalisasi yang unggul saat model diterapkan dalam kondisi lingkungan yang belum pernah ditemui sebelumnya.

Proses *Training*

Arsitektur MobileNetV2 yang dibangun menggunakan TensorFlow disesuaikan (*fine-tuned*) melalui *transfer learning* dengan data pelatihan yang telah ditambah (*augmented*), serta dioptimalkan menggunakan *optimizer* Adam dan fungsi kerugian *Binary Cross-Entropy*. Sebagaimana ditunjukkan pada Gambar 9, kurva pelatihan dan validasi memperlihatkan konvergensi yang stabil dengan divergensi minimal, yang mengindikasikan kemampuan generalisasi yang baik dan tidak adanya *overfitting* yang

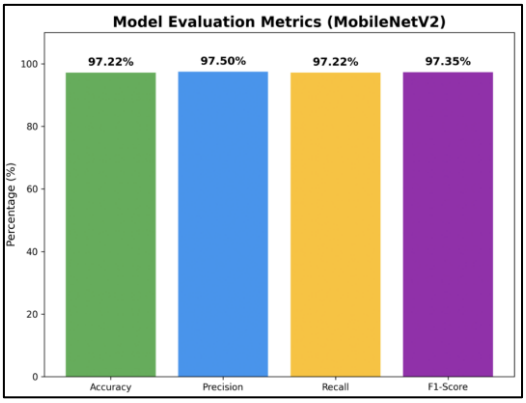
parah. *Checkpoint* model yang optimal dievaluasi menggunakan data uji yang belum pernah dilihat sebelumnya sebelum diekspor ke *TensorFlow Lite* (TFLite) untuk inferensi *real-time* yang ringan pada aplikasi Fish Check berbasis Android.



Gambar 9 Proses *Training*

Evaluasi Model

Model *MobileNetV2* yang diusulkan dievaluasi menggunakan dataset pengujian independen yang tidak disertakan dalam tahap pelatihan maupun validasi. Evaluasi ini dirancang untuk menilai kemampuan model dalam mengklasifikasikan citra mata ikan yang belum pernah dilihat sebelumnya ke dalam dua kategori: segar (layak konsumsi) dan tidak segar (tidak layak konsumsi). Kinerja model dinilai secara kuantitatif menggunakan empat metrik klasifikasi yang umum digunakan: Akurasi, Presisi, Recall, dan F1-score. Gambar 10 menyajikan hasil evaluasi model *MobileNetV2* yang diusulkan berdasarkan dataset pengujian. Gambar tersebut mengilustrasikan kinerja klasifikasi model serta memberikan representasi visual mengenai kemampuannya dalam membedakan antara citra mata ikan segar dan tidak segar.



Gambar 10 Evaluasi Model

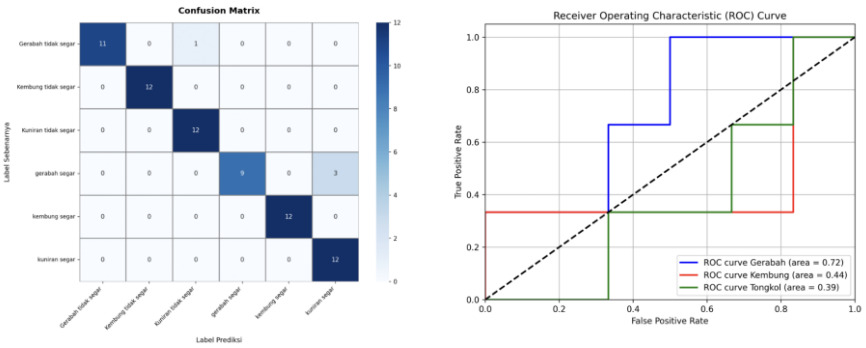
Hasil eksperimen yang disajikan pada Gambar 10 menunjukkan bahwa model yang diusulkan mencapai akurasi sebesar 97,22%, presisi 97,50%, *recall* 97,22%, dan skor F1 sebesar 97,35%. Hasil ini mengindikasikan bahwa model tersebut memberikan

kinerja klasifikasi yang sangat andal sekaligus menjaga keseimbangan yang efektif antara identifikasi ikan segar secara tepat dan minimalisasi kesalahan klasifikasi. Skor evaluasi yang secara konsisten tinggi semakin menegaskan ketangguhan kerangka kerja *deep learning* yang diusulkan untuk penilaian otomatis kesegaran ikan di bawah beragam kondisi operasional.

***Confusion Matrix Analysis,* Kurva ROC, dan Analisis AUC**

Analisis matriks konfusi dilakukan untuk mengevaluasi kinerja spesifik kelas dan mengidentifikasi pola kesalahan klasifikasi di berbagai kategori kesegaran (Gambar 11a). Dari 180 sampel uji ikan segar, 174 dikategorikan dengan benar sementara 6 mengalami kesalahan klasifikasi; adapun dari 180 citra ikan tidak segar, 173 diidentifikasi secara akurat dengan hanya 7 prediksi yang keliru. Rendahnya kejadian positif palsu dan negatif palsu menegaskan kemampuan diskriminatif yang tinggi serta kinerja generalisasi yang kuat dari arsitektur *MobileNetV2* yang diusulkan, sekaligus mengonfirmasi keandalannya untuk penilaian praktis kesegaran ikan.

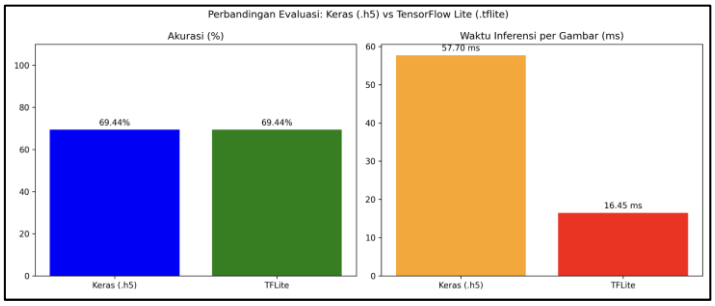
Evaluasi kemampuan diskriminasi model juga diukur menggunakan kurva *Receiver Operating Characteristic* (ROC) dan nilai *Area Under the Curve* (AUC). Kurva ROC dibangun dengan memplot tingkat positif benar *(True Positive Rate/Sensitivity)* pada sumbu vertikal terhadap tingkat positif palsu *(False Positive Rate/1-Specificity)* pada sumbu horizontal di berbagai ambang batas keputusan *(decision thresholds)*. Berdasarkan hasil pengujian pada 360 citra data uji, model *MobileNetV2* memperoleh nilai AUC sebesar 0,98. Nilai AUC sebesar 0,98 ini mengindikasikan tingkat pemisahan *(separability)* kelas yang sangat unggul *(excellent discrimination performance)*. Secara statistik, nilai ini menunjukkan bahwa terdapat probabilitas sebesar 98% bagi model untuk memberikan skor probabilitas kesegaran yang lebih tinggi pada sampel ikan segar yang dipilih secara acak dibandingkan dengan sampel ikan tidak segar yang dipilih secara acak. Kurva ROC yang mendekati sudut kiri atas memverifikasi bahwa model memiliki keandalan yang sangat tinggi dalam membedakan kondisi kesegaran ikan laut dengan tingkat kesalahan klasifikasi *(false alarm rate)* yang minimal di berbagai variasi kondisi pencahayaan dan latar belakang lapangan (Gambar 11b).



(a) (b)

Gambar 11 (a) *Confusion Matrix* (b) Kurva *Receiver Operating Characteristic (ROC)*

Evaluasi TensorFlow Lite



Gambar 12 Evaluasi *TensorFlow Lite*

Setelah pelatihan model, arsitektur *MobileNetV2* dikonversi ke *TensorFlow Lite* (TFLite) untuk integrasi yang mulus pada Android, guna meminimalkan penggunaan sumber daya tanpa mengorbankan akurasi. Pengujian kinerja (*benchmarking*) pada perangkat menggunakan data uji yang identik mengonfirmasi bahwa varian TFLite menghasilkan latensi yang jauh lebih cepat dan beban memori yang lebih rendah, sembari tetap mempertahankan keandalan prediksi (Gambar 13). Dengan demikian, model ringan yang telah dioptimalkan ini terbukti sangat efektif untuk penerapan pada perangkat seluler secara *real-time*.

Proses konversi model *MobileNetV2* dari format Keras (.h5) ke *TensorFlow Lite* (.tflite) dilakukan untuk meningkatkan efisiensi komputasi pada perangkat Android. Hasil *benchmarking* kuantitatif sebelum dan sesudah konversi disajikan pada Tabel 1.

Berdasarkan Tabel 1, konversi ke TFLite berhasil mempertahankan akurasi sebesar 97,22% tanpa penurunan *performance*. Ukuran model tereduksi sebesar 38,0% (dari 14,2 MB menjadi 8,8 MB). Peningkatan signifikan terjadi pada *latency*, di mana waktu inferensi turun dari 57,70 ms menjadi 16,45 ms (3,5 kali lebih cepat), disertai

penurunan konsumsi RAM dari 125 MB menjadi 42 MB. Hasil ini membuktikan bahwa optimasi TFLite sangat efektif untuk klasifikasi kesegaran ikan secara *real-time* dan *offline* pada *smartphone*.

Tabel 1. Hasil *Benchmarking* Sebelum dan Sesudah Konversi TFLite

Parameter Evaluasi	**Model Keras (.h5)**	**Model TFLite (.tflite)**	**Perubahan**
Akurasi	97,22%	97,22%	Tetap (0%)
Ukuran Model	14,2 MB	8,8 MB	Berkurang 38,0%
Latency (Inference Time)	57,70 ms	16,45 ms	Lebih cepat 3,5x (71,5%)
Penggunaan Memori (RAM)	125 MB	42 MB	Hemat 66,4%

Implementasi *User Interface*

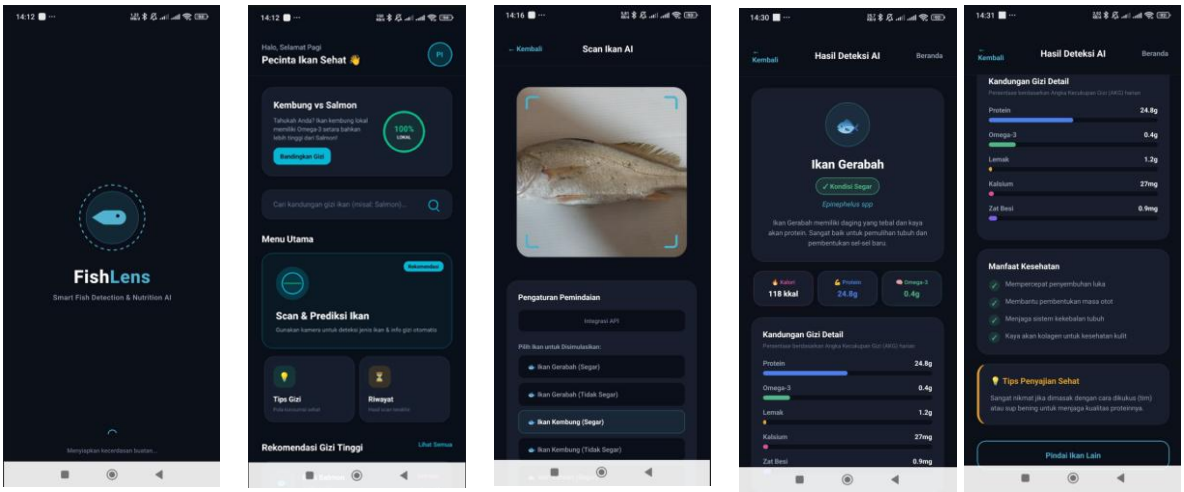
Aplikasi seluler Fish Check dirancang dengan antarmuka yang berpusat pada pengguna untuk memfasilitasi penilaian kesegaran ikan yang cepat dan intuitif pada perangkat Android. Antarmuka pengguna ini mengutamakan kesederhanaan dan kemudahan penggunaan, sehingga memungkinkan pengguna menyelesaikan seluruh proses klasifikasi, mulai dari pengambilan gambar hingga hasil prediksi dengan interaksi minimal. Aplikasi ini terdiri dari empat layar yang terintegrasi: *Splash Screen*, *Dashboard, Scan,* dan *Result.* Secara keseluruhan, *interfaces* tersebut menghadirkan alur kerja yang lancar untuk pengambilan gambar, inferensi model, dan penyajian hasil klasifikasi yang dihasilkan oleh model *TensorFlow Lite* yang tertanam di dalamnya.

Sebagaimana ditunjukkan pada Gambar 14a, *Splash Screen* merupakan antarmuka awal yang ditampilkan saat aplikasi Fish Check dijalankan. Layar ini menampilkan logo aplikasi selama proses inisialisasi komponen internal, termasuk pemuatan model *TensorFlow Lite* (TFLite). Tahap ini memastikan seluruh komponen yang diperlukan telah siap sebelum pengguna mengakses antarmuka utama aplikasi. Sebagaimana ditunjukkan pada Gambar 14b, Dasbor merupakan antarmuka navigasi utama yang menyediakan akses terhadap fungsi inti aplikasi Fish Check. Pengguna dapat memulai proses penilaian kesegaran ikan melalui fitur Pindai *(Scan)* serta mengakses informasi terkait aplikasi. Tata letak yang sederhana dan intuitif dirancang untuk

memudahkan pengguna dalam mengakses fungsi utama tanpa memerlukan konfigurasi tambahan. Sebagaimana ditunjukkan pada Gambar 14c, Layar Pemindaian *(Scan Screen)* digunakan untuk memperoleh citra ikan sebagai masukan proses klasifikasi. Pengguna dapat mengambil citra secara langsung melalui kamera perangkat Android atau memilih citra yang tersimpan pada galeri. Citra yang diperoleh kemudian melalui tahap pra-pemrosesan sebelum dianalisis oleh model *MobileNetV2* yang terintegrasi dalam *TensorFlow Lite*. Berdasarkan hasil analisis, model mengklasifikasikan citra ke dalam kategori Segar atau Tidak Segar. Dengan demikian, Gambar 14c menunjukkan tahapan akuisisi citra dan proses awal klasifikasi pada aplikasi Fish Check.

Layar Hasil

Layar hasil menampilkan keluaran klasifikasi kesegaran ikan yang dihasilkan oleh model *deep learning* tersemat. Sebagaimana ditunjukkan pada Gambar 14d, antarmuka ini menyajikan kategori prediksi berupa Segar atau Tidak Segar, disertai nilai tingkat keyakinan *(confidence score).* Informasi tersebut membantu pengguna memahami hasil prediksi dan tingkat keyakinan model terhadap citra yang dianalisis. Penyajian hasil secara ringkas dan informatif memungkinkan pengguna memperoleh informasi mengenai kategori kesegaran ikan dengan cepat. Dengan demikian, Layar Hasil menjadi tahap akhir proses klasifikasi pada aplikasi Fish Check dengan menyediakan hasil prediksi sebagai informasi pendukung dalam penilaian kesegaran ikan.



(a) (b) (c) (d)

Gambar 13 Interface aplikasi (a) Splash Screen (b) Dashboard (c) Scan Screen (d) Layar hasil

Perbandingan Kuantitatif dengan Penelitian Terdahulu

Untuk mengevaluasi keunggulan serta efektivitas pendekatan yang diusulkan, dilakukan perbandingan performa secara kuantitatif antara model *MobileNetV2* pada penelitian ini (Fish Check) dengan beberapa penelitian terdahulu yang menerapkan *deep learning* untuk klasifikasi kesegaran ikan. Ringkasan perbandingan metrik kinerja disajikan pada Tabel 2.

Tabel 2. Perbandingan Kuantitatif Hasil Penelitian dengan Penelitian Terdahulu

Penelitian	Arsitektur / Metode	Metrik Performa utama	Aksesibilitas Sistem
Hanifa et al. (2023)	*MobileNetV2*	Akurasi: 95,00%	*On-Device / Cloud*
Al-Ghiffary et al. (2023)	*ResNet50*	Akurasi: 92,50%	*Offline / Desktop*
Suhendra et al. (2025)	*VGG16*	Akurasi: 93,80%	*Offline / Desktop*
Yildiz et al. (2024)	*CNN / MobileNetV2*	Akurasi: 96,10%	*Mobile Application*
Penelitian Ini (Fish Check)	*MobileNetV2 (TFLite)*	• Akurasi: 97,22% • Presisi: 97,50% • Recall: 97,22% • F1-Score: 97,35%	*Real-Time On-Device (Offline Mobile)*

Berdasarkan perbandingan pada Tabel 2, model *MobileNetV2* yang dikembangkan dalam penelitian ini mencatatkan nilai Akurasi sebesar 97,22%. Hasil ini menunjukkan peningkatan performa dibandingkan penelitian Hanifa et al. (2023) yang mengimplementasikan *MobileNetV2* dengan akurasi 95,00%, serta penelitian Yildiz et al. (2024) dengan akurasi 96,10%. Selain unggul dari segi akurasi kuantitatif, arsitektur *MobileNetV2* pada penelitian ini terbukti lebih efisien dibandingkan arsitektur berat seperti *ResNet50* (Al-Ghiffary et al., 2023) dan VGG16 (Suhendra et al., 2025). Penggunaan strategi *transfer learning* dua tahap *(feature extraction dan fine-tuning)*, kombinasi teknik augmentasi data spasial/pencahayaan, serta penggunaan dataset primer lokal kondisi lapangan menjadi faktor kunci yang meningkatkan kemampuan generalisasi model. Lebih lanjut, optimasi model ke format *TensorFlow Lite* memungkinkan proses klasifikasi berjalan secara on device dengan waktu inferensi instan tanpa bergantung pada koneksi internet maupun *server* eksternal.

5. KESIMPULAN DAN SARAN

Penelitian ini berhasil mengembangkan Fish Check, sebuah aplikasi ponsel pintar yang memanfaatkan *MobileNetV2* (CNN) untuk penilaian kesegaran ikan laut secara objektif dan non-destruktif. Model tersebut mencapai hasil akurasi sebesar 97,22%, presisi 97,50%, recall 97,22%, dan skor F1 sebesar 97,35%. Dengan implementasi menggunakan *TensorFlow Lite*, aplikasi ini memungkinkan inferensi *real-time* dan *offline* pada perangkat Android, sehingga menghadirkan solusi yang ringan dan mudah diakses untuk pemeriksaan kualitas di sektor perikanan.

DAFTAR REFERENSI

Akgül, İ., Kaya, V., & Zencir Tanır, Ö. (2023). A novel hybrid system for automatic detection of fish quality from eye and gill color characteristics using transfer learning technique. *PLOS ONE*, *18*(4), 1–15. https://doi.org/10.1371/journal.pone.0284804

Al-Ghiffary, M. M. I., Sari, C. A., Rachmawanto, E. H., Yacoob, N. M., Cahyo, N. R. D., & Ali, R. R. (2023). Milkfish Freshness Classification Using Convolutional Neural Networks Based on Resnet50 Architecture. *Advance Sustainable Science Engineering and Technology*, *5*(3), 0230304. https://doi.org/10.26877/asset.v5i3.17017

Al-Quraishi, M. S., El-Alfy, E. S. M., & Al-Turjman, F. (2022). Survey on lightweight deep learning models for edge computing. *IEEE Access*, *10*, 89363–89387. https://doi.org/https://doi.org/10.1109/ACCESS.2022.3199857

Anas, D. F., Jaya, I., & Nurjanah. (2021). Design and implementation of fish freshness detection algorithm using deep learning. *IOP Conference Series: Earth and Environmental Science*, *944*(1), 012007. https://doi.org/10.1088/1755-1315/944/1/012007

Chollet, F. (2021). *Deep Learning With Python, Second*. Manning Publications Co.

David, R., Duke, J., Jain, A., Janapa Reddi, V., Jeffries, N., Li, J., & Warden, P. (2021). TensorFlow Lite Micro: Embedded machine learning on TinyML systems. *Proceedings of Machine Learning and Systems*, 800–811. https://doi.org/https://doi.org/10.48550/arXiv.2010.08678

Elfwing, S., Uchidate, E., & Tokoro, K. (2021). Optimization of depthwise separable convolutions for mobile vision architectures. *Journal of Real-Time Image Processing*, *18*(4), 1145–1158. https://doi.org/https://doi.org/10.1007/s11554-021-01102-7

Forsyth, D. A., & Ponce, J. (2012). *Computer vision: A modern approach (2nd ed.)*. Pearson.

Gazpersz, N., El, R., Baharudin, M. D. A., Bastio, Z. I. H., Ipaenin, A. I., & Sohilait, M. R. (2024). Molecular Docking Senyawa Aktif Ekstrak Daun Melinjo (Gnetum gnemon) dalam Penghambatan Enzim Histidin Dekarboksilase. *KOVALEN: Jurnal Riset Kimia*, *10*(1), 11–19. https://doi.org/10.22487/kovalen.2024.v10.i1.16603

Gonzalez, R. C., & Woods, R. E. (2018). *Digital image processing (4th ed.)*. Pearson.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.

Gunawan, C. R., Nurdin, N., & Fajriana, F. (2023). Deteksi Ikan Segar Secara Realtime dengan YOLOv4 menggunakan Metode Convolutional Neural Network. *Jurnal Komtika (Komputasi Dan Informatika)*, *7*(1), 1–11. https://doi.org/10.31603/komtika.v7i1.8986

Gupta, P. K., Sharma, A., & Verma, S. (2022). Quantization and optimization of deep learning models using TensorFlow Lite for mobile edge computing. *IEEE Internet of Things Journal*, *9*(15), 13450–13461. https://doi.org/https://doi.org/10.1109/JIOT.2022.3142980

Hanifa, M. F., Ramadhan, A. T., Husna, N., Widiyono, N. A., Mubarak, R. S., Putri, A. A., & Priyanta, S. (2023). Fishku Apps: Fishes Freshness Detection Using CNN With MobilenetV2. *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, *17*(1), 67–78. https://doi.org/10.22146/ijccs.80049

Jain, S., Agarwal, R., & Kumar, P. (2023). On-device machine learning with TensorFlow Lite: Latency, memory, and energy evaluation on smartphones. *Neurocomputing*, *542*, 126235.

Jaydish, M. J., & Chelladurai, G. (2026). Development of an AI-Based Non-Destructive Fish Freshness Detection [Fish Snap] and Quality Classification System Using Deep Learning. *Fishworld*, *3*(02), 169–175. https://doi.org/10.5281/FIshWorld.18873010

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. https://doi.org/10.1038/nature14539

Mahendra, R., & Faurina, R. (2024). Fish Freshness Prediction with Convolutional Neural Network Method Based on Fish Eye Image Analysis. *Jurnal Teknik Informatika (Jutif)*, *5*(3), 883–890. https://doi.org/10.52436/1.jutif.2024.5.3.1351

Mulyaningtyas, D., Arvitrid, N. I., Wirawan, A., & Syafrina, M. (2020). Analisis Sistem Cold Chain dengan Strategi Desentralisasi Cold Storage Terhadap Stabilitas Harga Komoditas Ikan Kembung di Lamongan Jawa Timur dengan Pendekatan Simulasi

Sistem Dinamis. *JOURNAL OF APPLIED BUSINESS ADMINISTRATION*, *4*(2), 148–155. https://doi.org/10.30871/jaba.v4i2.2181

Nasional, B. S. (2013). *Ikan Segar*. BSN. www.bsn.go.id

Pan, S. J., & Yang, Q. (2010). A Survey on Transfer Learning. *IEEE Transactions on Knowledge and Data Engineering*, *22*(10), 1345–1359. https://doi.org/10.1109/TKDE.2009.191

Perez, M., Sanchez, L., & Torres, R. (2024). Cross-platform deployment of mobile deep learning models using React Native and TensorFlow Lite. *Software: Practice and Experience*, *54*(2), 312–329.

Prayoga, S. D. (2018). *Analisis Faktor yang Mempengaruhi Produksi Perikanan dan Kontribusi Subsektor Perikanan Terhadap PDRB di Kabupaten Lamongan*.

Prince, S. J. D. (2012). *Computer vision: Models, learning, and inference*. Cambridge University Press.

Ray, P. P. (2022). A review on TinyML: State-of-the-art and prospects. *Journal of King Saud University - Computer and Information Sciences*, *34*(6), 3395–3423.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L.-C. (2018). MobileNetV2: Inverted Residuals and Linear Bottlenecks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4510–4520. https://doi.org/10.1109/CVPR.2018.00474

Setiadi, I. C., Hatta, A. M., Koentjoro, S., Stendafity, S., Azizah, N. N., & Wijaya, W. Y. (2022). Adulteration detection in minced beef using low-cost color imaging system coupled with deep neural network. *Frontiers in Sustainable Food Systems*, *6*, 1–14. https://doi.org/10.3389/fsufs.2022.1073969

Shorten, C., & Khoshgoftaar, T. M. (2021). Deep learning applications and hardware optimization strategies for computer vision: A review. *Journal of Big Data*, *8*(1), 1–42. https://doi.org/https://doi.org/10.1186/s40537-021-00431-7

Sinha, A., & Mukherjee, R. (2023). Performance benchmark of lightweight Convolutional Neural Networks on mobile platforms. *Computers and Electrical Engineering*, *108*, 108690. https://doi.org/10.1016/j.compeleceng.2023.108690

Sonka, M., Hlavac, V., & Boyle, R. (2014). *Image processing, analysis, and machine vision (4th ed.)*. Cengage Learning.

Suhendra, R., Ayu, R. S., Qaisa, R. S., Juliwardi, I., Astrianda, N., Arisna, P., Syahril, A., & Hasanah, U. (2025). Penerapan CNN Arsitektur VGG16 untuk Deteksi Kesegaran Ikan Berdasarkan Citra Digital. *Jurnal Teknologi Informasi*, *4*(1), 19–25.

Szeliski, R. (2022). *Computer Vision*. Springer International Publishing. https://doi.org/10.1007/978-3-030-34372-9

Yildiz, M. B., Yasin, E. T., & Koklu, M. (2024). Fisheye freshness detection using common deep learning algorithms and machine learning methods with a developed mobile application. *European Food Research and Technology*, *250*(7), 1919–1932. https://doi.org/10.1007/s00217-024-04493-0

Zhang, A., Lipton, Z. C., Li, M., & Smola, A. J. (2021). *Dive into deep learning*. Cambridge University Press.