

Klasifikasi *Brand* Sepatu Nike Berbasis Citra Dengan Algoritma *Convolution Neural Network* (Cnn)

Sulthan Saalim Rabbani Atmadja^{1*}, Aris Haris Rismayana²

^{1,2}Program Studi Teknik Informatika, Politeknik TEDC Bandung

Email: ¹altanslimmm@gmail.com, ²rismayana@poltektedc.ac.id

INFORMASI ARTIKEL

Histori artikel:

Naskah masuk, 15 Juli 2025

Direvisi, 19 Juli 2025

Diiterima, 28 Juli 2025

Kata kunci :

Convolutional Neural Network

Klasifikasi

Sepatu Nike Original dan KW

Deep Learning

Pre-Processing

ABSTRAK

Abstract- *The widespread circulation of counterfeit (fake) Nike shoes has made it increasingly difficult to distinguish between genuine and fake products due to their high visual similarity. This study aims to develop a Nike shoe classification system based on digital image processing using a quantitative approach and experimental methods. The model employed is a Convolutional Neural Network (CNN) based on the MobileNetV2 architecture, utilizing transfer learning. The dataset consists of 200 images of Nike shoes categorized into two classes (original and fake), which have undergone manual labeling and image augmentation. The model was trained using TensorFlow and Keras, and evaluated based on accuracy, precision, recall, and the confusion matrix. The evaluation results show that MobileNetV2 achieved 96% accuracy on the test data, with a recall of 0.94 for the fake class and 0.99 for the original class. Compared to VGG16 as a benchmark model, MobileNetV2 proved to be lighter, faster, and more balanced in classification performance. This study demonstrates that the deep learning approach using MobileNetV2 is effective for detecting counterfeit shoes and has the potential to be implemented in mobile applications as an automatic product authentication solution. However, the model still faces challenges in classification accuracy under varying lighting conditions and image angles, as well as limitations in dataset size.*

Abstrak- Peredaran sepatu Nike tiruan (KW) yang semakin luas menimbulkan kesulitan dalam membedakan produk asli dan palsu karena kemiripan visual yang tinggi. Penelitian ini bertujuan membangun sistem klasifikasi sepatu Nike berbasis citra digital menggunakan pendekatan kuantitatif dengan metode eksperimen. Model yang digunakan adalah *Convolutional Neural Network* (CNN) berbasis arsitektur *MobileNetV2*, dengan pendekatan *transfer learning*. *Dataset* terdiri atas 200 gambar sepatu Nike yang dibagi menjadi dua kelas (original dan KW), dan telah melalui proses augmentasi serta pelabelan manual. Model dilatih menggunakan *TensorFlow* dan *Keras*, kemudian dievaluasi berdasarkan akurasi, *precision*, *recall*, dan *confusion matrix*. Hasil evaluasi menunjukkan bahwa *MobileNetV2* mencapai akurasi 96% pada data uji, dengan *recall* 0.94 untuk kelas KW dan 0.99 untuk kelas original. Dibandingkan dengan VGG16 sebagai model pembandingan, *MobileNetV2* terbukti lebih ringan, cepat, dan seimbang dalam performa klasifikasi. Penelitian ini menunjukkan bahwa pendekatan *deep learning* dengan *MobileNetV2* efektif untuk mendeteksi sepatu palsu, serta berpotensi diterapkan dalam aplikasi *mobile* sebagai solusi autentikasi produk secara otomatis. Namun, penelitian ini masih menghadapi tantangan dalam akurasi klasifikasi pada kondisi pencahayaan dan sudut pengambilan gambar yang bervariasi, serta keterbatasan jumlah data yang digunakan.

Copyright © 2025 LPPM - STMIK IKMI Cirebon
This is an open access article under the CC-BY license

Penulis Korespondensi:

Sulthan Saalim Rabbani Atmadja

Program Studi Teknik Informatika,

Politeknik TEDC Bandung

Jl. Jl. Politeknik Pesantren KM2 Cibabat Cimahi Utara - Cimahi - Jawa Barat - Indonesia

Email: altanslimmm@gmail.com

1. Pendahuluan

Industri sepatu telah berkembang pesat dalam dua dekade terakhir, baik dari segi teknologi produksi, desain, maupun strategi pemasaran digital. Sepatu kini tidak hanya berfungsi sebagai pelindung kaki, tetapi juga menjadi simbol gaya hidup dan status sosial. Nike merupakan salah satu merek terdepan yang dikenal karena kemampuannya menggabungkan inovasi teknologi dengan desain ikonik, serta memperkuat identitas melalui logo “swoosh” dan slogan “Just Do It”.

Namun, tingginya popularitas Nike juga memicu peredaran sepatu tiruan (KW) yang secara visual sangat mirip dengan produk asli. Hal ini menimbulkan kerugian bagi produsen dan distributor resmi, serta membingungkan konsumen. Sepatu KW sering kali sulit dibedakan hanya melalui pengamatan visual biasa, sehingga dibutuhkan pendekatan berbasis teknologi untuk mengidentifikasi keaslian produk secara lebih akurat.

Dalam studi yang dilakukan oleh Panchal dan Vora (2024), disebutkan bahwa peredaran sepatu Nike palsu kini sangat marak, dan teknik deteksi berbasis visual konvensional tidak lagi efektif. Mereka mengembangkan sistem klasifikasi dengan model YOLOv8 yang mampu mendeteksi perbedaan antara sepatu asli dan palsu dengan tingkat mAP mencapai 95%, disertai presisi dan recall di atas 90% [1]. Demikian pula, Garcia-Cotte et al. (2024) menunjukkan bahwa model *deep learning* berbasis citra, tanpa perlu perangkat keras tambahan atau tag fisik, dapat mencapai akurasi hingga 99,71% hanya melalui pengambilan gambar dengan smartphone dalam kondisi alami [2]. Namun, di Indonesia, pendekatan teknologi semacam ini masih sangat minim, khususnya dalam konteks klasifikasi keaslian sepatu merek tertentu seperti Nike. Studi yang dilakukan oleh Renaldi dan Lubis (2024) mencoba mengklasifikasikan model sepatu Nike berdasarkan gambar menggunakan arsitektur *MobileNetV2*, namun akurasi yang dicapai hanya 66,67% [3]. Hal ini menandakan bahwa masih terdapat celah yang cukup besar untuk penelitian lanjutan dalam mengembangkan sistem klasifikasi keaslian sepatu berbasis citra yang lebih akurat dan praktis.

Kemajuan kecerdasan buatan, khususnya dalam pengolahan citra dan *deep learning*, membuka peluang untuk mengembangkan sistem

deteksi otomatis produk KW. *Convolutional Neural Network* (CNN) dikenal efektif dalam mengenali pola visual kompleks seperti tekstur, bentuk, dan warna, serta banyak digunakan dalam berbagai aplikasi, mulai dari pengenalan wajah hingga klasifikasi objek.

Namun, penerapan CNN untuk klasifikasi keaslian sepatu, khususnya merek Nike, masih jarang dilakukan di Indonesia. Penelitian ini bertujuan mengembangkan model CNN untuk membedakan sepatu Nike original dan tiruan secara efisien. Fokus penelitian mencakup pemilihan arsitektur CNN yang optimal, teknik augmentasi citra, dan evaluasi menggunakan metrik akurasi, presisi, dan *recall*. Diharapkan, hasilnya dapat menjadi solusi praktis untuk membantu konsumen membuat keputusan pembelian yang lebih tepat di tengah maraknya produk tiruan. Kebaruan (*novelty*) dari penelitian ini terletak pada penerapan arsitektur *MobileNetV2* secara langsung dalam klasifikasi keaslian sepatu Nike berbasis citra tanpa *preprocessing* eksplisit, serta integrasi model ke dalam aplikasi mobile dengan pengujian konteks dunia nyata

2. Penelitian Terkait

2.1. Brand Sepatu Nike

Nike merupakan *brand* global yang mendominasi industri alas kaki lewat inovasi, strategi pemasaran agresif, dan citra merek yang kuat. Elemen visual seperti logo “swoosh” dan warna khas menjadikannya mudah dikenali. Namun, popularitasnya juga memicu maraknya produk tiruan yang meniru desain dan identitas visualnya. Menurut Garcia-Cotte et al. (2024), pemalsuan produk berbasis visual kini semakin canggih dan sulit dikenali oleh konsumen awam karena desain dan teksturnya mendekati produk asli [2].

Kondisi ini menyebabkan banyak konsumen tidak sadar telah membeli barang palsu karena tidak mampu membedakan secara kasat mata. Di Indonesia, *platform* edukatif seperti SiBersih.id menjelaskan bahwa banyak sepatu KW dibuat dengan detail sangat mirip, mulai dari jahitan, bentuk sol, hingga kode produksi, yang secara teori bisa dibedakan dengan pelatihan visual berbasis teknologi [4]. Oleh karena itu, diperlukan pendekatan otomatis berbasis citra untuk mengidentifikasi ciri khas produk asli secara lebih objektif dan cepat.

2.2. Pengolahan Citra

Pengolahan citra digital merupakan bidang ilmu komputer yang menangani analisis gambar

untuk memperoleh informasi penting. Dalam klasifikasi sepatu Nike, teknik ini digunakan untuk mengekstrak fitur visual seperti *outsole*, tekstur, logo, dan warna. Proses awal seperti *resize*, konversi ke *grayscale*, normalisasi piksel, serta augmentasi data (*flipping*, rotasi, *zooming*) menjadi tahap krusial sebelum gambar digunakan dalam pelatihan model. Studi oleh Hermana, Husada, dan Kurniawan (2024) menunjukkan bahwa teknik *balancing* data menggunakan SMOTE berhasil meningkatkan performa model CNN untuk klasifikasi sepatu Converse palsu [5]. Di sisi lain, Renaldi dan Lubis (2024) juga memanfaatkan teknik *preprocessing* sederhana pada *dataset* sepatu Nike, namun keterbatasan dalam variasi data menjadi hambatan utama dalam mencapai akurasi optimal [3]. Karena itu, strategi *preprocessing* yang tepat penting agar data citra yang masuk ke CNN dapat merepresentasikan perbedaan visual antara sepatu ORI dan KW secara akurat.

2.3. Deep learning

Deep learning merupakan salah satu cabang dari *machine learning* yang secara khusus memanfaatkan jaringan saraf tiruan (*artificial neural networks*) dengan banyak lapisan (*deep architecture*) untuk mempelajari representasi data secara otomatis. Tidak seperti metode *machine learning* tradisional yang mengandalkan ekstraksi fitur manual, *deep learning* dapat secara otomatis mempelajari fitur penting langsung dari data mentah seperti gambar, teks, atau suara [6].

Struktur *deep learning* terdiri dari banyak lapisan tersembunyi yang secara bertahap mengubah data menjadi representasi yang lebih abstrak. Lapisan awal menangkap fitur dasar seperti tepi dan warna, sedangkan lapisan dalam mengenali pola kompleks seperti bentuk dan objek [7]. *Deep learning* digunakan luas karena mampu mengenali pola *non-linear* dan mengolah data berdimensi tinggi secara efektif.

2.4. Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) adalah CNN adalah arsitektur *deep learning* yang dirancang khusus untuk memproses data dua dimensi seperti gambar. Dengan lapisan *convolutional*, *pooling*, dan *fully connected*, CNN mengekstraksi fitur spasial secara bertahap. Popularitasnya meningkat sejak keberhasilan AlexNet (Krizhevsky et al., 2012) yang secara signifikan meningkatkan akurasi dalam kompetisi *ImageNet* [8]. Di Indonesia, berbagai studi menunjukkan keunggulan CNN dibanding algoritma konvensional. Penelitian Gina Annisa dan Ninuk Wiliani (2025) membuktikan bahwa CNN menghasilkan akurasi klasifikasi sepatu yang lebih tinggi dibanding SVM pada dataset lokal [9]. Selain itu, jurnal oleh Komunitas Ilmiah Teknologi

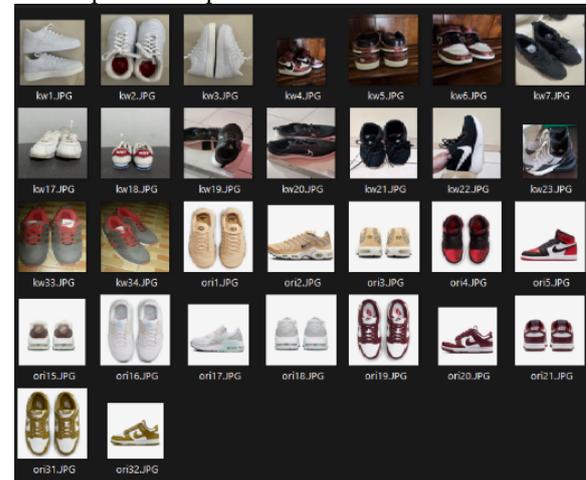
Informasi Indonesia (2024) memperlihatkan bahwa CNN memiliki akurasi lebih tinggi pada *dataset* CIFAR-10 dibandingkan SVM dan ANN [10]. Penelitian-penelitian tersebut membuktikan bahwa CNN merupakan pilihan tepat untuk sistem klasifikasi citra, termasuk dalam membedakan sepatu Nike ORI dan KW.

3. Metode

3.1. Dataset

Dataset berisi 200 gambar sepatu Nike, masing-masing 100 untuk kelas ORI dan KW, diperoleh dari *platform open-source* Kaggle. Gambar mencerminkan variasi jenis sepatu, sudut pandang, latar, pencahayaan, dan posisi untuk merepresentasikan kondisi nyata. Seluruh gambar diperiksa manual sebelum pelatihan untuk memastikan kualitas dan keberagaman data.

Visualisasi contoh citra dalam *dataset* kategori KW dapat dilihat pada Gambar 1 berikut:



Gambar 1. Contoh variasi visual gambar sepatu

Penelitian ini tidak menerapkan *preprocessing* eksplisit seperti *grayscale*, *histogram*, atau *filtering*. Gambar digunakan dalam format RGB asli dari sumber *dataset*, sementara augmentasi dan normalisasi piksel dilakukan otomatis melalui *layer* bawaan *TensorFlow*, yang dijelaskan pada subbab berikutnya.

3.2. Arsitektur Model

Model klasifikasi ini menggunakan *transfer learning* dengan *MobileNetV2* yang telah dilatih pada *ImageNet*. Arsitektur ini dipilih karena ringan, efisien, dan tetap unggul dalam klasifikasi citra. Bobotnya dibekukan (*trainable=False*) untuk mempercepat pelatihan dan mencegah *overfitting*, mengingat data yang terbatas. Sebelum masuk ke *MobileNetV2*, gambar diproses melalui lapisan augmentasi otomatis menggunakan *tf.keras.Sequential*, yang terdiri dari:

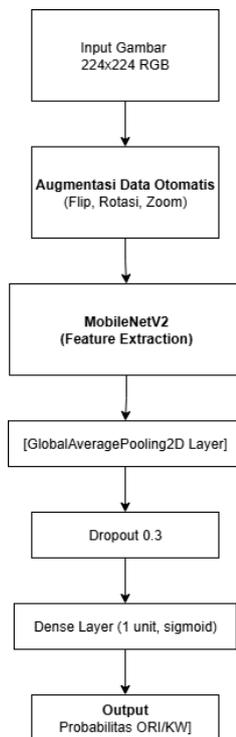
- RandomFlip('horizontal')*: membalik gambar secara *horizontal*,

- b. *RandomRotation*(0.1): merotasi gambar secara acak sebesar $\pm 10\%$,
- c. *RandomZoom*(0.1): melakukan *zoom* acak sebesar $\pm 10\%$.

Augmentasi dilakukan secara *real-time* selama pelatihan, sehingga memperkaya variasi data tanpa menambah ukuran *dataset* secara eksplisit.

Setelah ekstraksi fitur oleh *MobileNetV2*, output diratakan dengan *GlobalAveragePooling2D*, diberi *Dropout* 0.3, lalu diproses oleh *Dense* beraktivasi *sigmoid* untuk menghasilkan probabilitas klasifikasi ORI atau KW.

Alur proses data dalam arsitektur model ini divisualisasikan pada Gambar 2 berikut:



Gambar 2. Diagram alur model CNN berbasis *MobileNetV2*

Model dilatih selama 10 *epoch* dengan *batch size* 32, menggunakan *optimizer* Adam (*learning rate* 0.001) dan fungsi *loss binary_crossentropy* untuk klasifikasi biner. Pelatihan dilakukan di *Python* dengan *TensorFlow* dan *Keras*, lalu dievaluasi menggunakan data uji yang telah dipisahkan.

3.3. Skema Pelatihan dan Evaluasi

Dataset dibagi menjadi dua bagian menggunakan metode *stratified shuffle split* untuk menjaga proporsi kelas yang seimbang di setiap subset. Rasio pembagian data adalah 80% untuk pelatihan (*training*) dan 20% untuk validasi (*validation*). Teknik *k-fold cross-validation* tidak diterapkan dalam penelitian ini karena jumlah *dataset* yang terbatas (200 gambar), sehingga

pengulangan pelatihan pada subset kecil dapat meningkatkan risiko *overfitting* serta memerlukan waktu pelatihan yang lebih panjang. Oleh karena itu, pendekatan *stratified hold-out split* dianggap lebih sesuai dalam konteks ini, karena tetap menjaga keseimbangan antar kelas dan memastikan model diuji pada data yang belum pernah dilihat selama pelatihan.

Proses pelatihan model dilakukan menggunakan *TensorFlow* dan *Keras*, dengan parameter pelatihan sebagai berikut: jumlah *epoch* sebanyak 10, *batch size* 32, *optimizer* Adam, *learning rate* 0.001, dan fungsi *loss binary_crossentropy*. Pelatihan dilakukan pada lingkungan pengembangan *Python* menggunakan GPU.

Evaluasi model dilakukan menggunakan metrik akurasi untuk menilai kemampuan model dalam mengklasifikasikan gambar secara biner. Selain itu, digunakan *confusion matrix* untuk melihat distribusi prediksi benar dan salah antara kelas ORI dan KW. Evaluasi dilakukan baik pada data uji yang dipisahkan sebelumnya, maupun pada hasil konversi model ke format *TensorFlow Lite* (.tflite) yang diuji pada perangkat Android. Hasil pengujian lebih lanjut, termasuk performa aplikasi mobile, akan dibahas secara lebih rinci pada Bab 4.

4. Hasil dan Pembahasan

4.1. Hasil Pelatihan Model

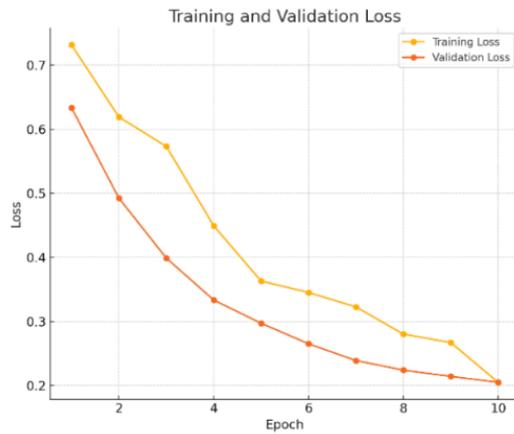
Model CNN berbasis *MobileNetV2* dilatih menggunakan 200 gambar dengan distribusi seimbang antara kelas ORI dan KW. Pelatihan berlangsung selama 10 *epoch* dengan *batch size* 32, menggunakan *optimizer* Adam (*learning rate* 0.001). Gambar diproses melalui augmentasi otomatis sebelum masuk ke jaringan *transfer learning*.



Gambar 3. Grafik akurasi model

Berdasarkan Gambar 3, grafik akurasi model terlihat bahwa baik akurasi pelatihan maupun validasi mengalami peningkatan signifikan dari

epoch ke-1 hingga ke-4, lalu stabil hingga epoch ke-10. Akurasi validasi mencapai sekitar 90%, dan nilainya tetap berada di atas akurasi pelatihan pada sebagian besar epoch, yang mengindikasikan bahwa model tidak mengalami *overfitting* secara signifikan.



Gambar 4. Grafik loss model

Berdasarkan Gambar 4, model nilai *loss* pada pelatihan dan validasi mengalami penurunan konsisten dari awal hingga akhir pelatihan. Nilai *loss* pelatihan menurun dari sekitar 0.73 menjadi di bawah 0.2, sementara *loss* validasi juga menunjukkan tren penurunan yang stabil dari sekitar 0.63 ke sekitar 0.21. Hal ini menunjukkan bahwa proses pembelajaran berjalan efektif dan model berhasil meminimalkan kesalahan secara bertahap.

Secara keseluruhan, model menunjukkan performa pelatihan yang baik dan stabil, serta siap untuk dievaluasi lebih lanjut menggunakan data uji.

4.2. Evaluasi Menggunakan Data Uji

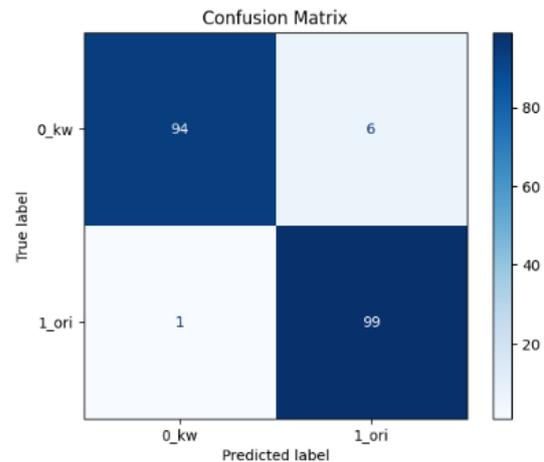
Setelah model selesai dilatih menggunakan *dataset* pelatihan dan validasi, evaluasi lebih lanjut dilakukan terhadap data uji untuk mengukur kemampuan generalisasi model dalam mengklasifikasikan citra sepatu ke dalam dua kategori, yaitu original (ORI) dan KW (imitasi). *Dataset* uji terdiri dari 200 gambar, masing-masing 100 gambar untuk kelas ORI dan 100 gambar untuk kelas KW.

Evaluasi dilakukan menggunakan metrik akurasi, serta *confusion matrix* untuk melihat distribusi klasifikasi benar dan salah antar kelas. Hasil *confusion matrix* disajikan pada Tabel 1 berikut:

Tabel 1. Confusion matrix model CNN pada data uji

| | Prediksi KW | Prediksi ORI |
|------------------|-------------|--------------|
| KW (sebenarnya) | 94 | 6 |
| ORI (sebenarnya) | 1 | 99 |

Visualisasi dari *confusion matrix* tersebut dapat dilihat pada Gambar 5 berikut:



Gambar 5. Visualisasi *confusion matrix* model CNN pada data uji

Selain itu, evaluasi juga dilakukan terhadap metrik *precision*, *recall*, dan *F1-score* untuk masing-masing kelas. Hasilnya dirangkum dalam Tabel 2 berikut:

Tabel 2. Metrik evaluasi model CNN pada data uji

| Kelas | <i>Precision</i> | <i>Recall</i> | <i>F1-score</i> | Support |
|-----------------|------------------|---------------|-----------------|---------|
| KW | 0.99 | 0.94 | 0.96 | 100 |
| ORI | 0.94 | 0.99 | 0.97 | 100 |
| Macro Avg | 0.97 | 0.96 | 0.96 | 200 |
| Weighted Avg | 0.97 | 0.96 | 0.96 | 200 |
| <i>Accuracy</i> | 0.96 | | | 200 |

Tabel 2 menunjukkan nilai *precision* menunjukkan bahwa dari seluruh prediksi positif yang dihasilkan model, sebagian besar merupakan prediksi yang benar. *Recall* menggambarkan kemampuan model dalam mendeteksi setiap kelas dengan baik, di mana kelas ORI memiliki *recall* 0.99 dan kelas KW sebesar 0.94. Nilai *F1-score* berada pada kisaran 0.96–0.97 yang menunjukkan keseimbangan antara *precision* dan *recall*, serta performa model yang stabil di kedua kelas.

Dengan akurasi keseluruhan sebesar 96%, model ini terbukti mampu mengklasifikasikan citra sepatu secara efektif, bahkan pada data yang belum pernah dilihat sebelumnya. Hasil ini juga menunjukkan bahwa penggunaan *MobileNetV2* sebagai arsitektur *transfer learning* cukup berhasil diterapkan dalam konteks klasifikasi sepatu original dan KW berbasis citra.

4.3. Implementasi Model & Pengujian pada Aplikasi Mobile

Setelah pelatihan dan pengujian di lingkungan pengembangan, model CNN dikonversi ke format *TensorFlow Lite* (.tflite) untuk diintegrasikan ke aplikasi Android menggunakan *TensorFlow Lite* API. Aplikasi memungkinkan input gambar dari galeri atau kamera dan menghasilkan output

klasifikasi (“KW” atau “ORI”) disertai nilai sigmoid.

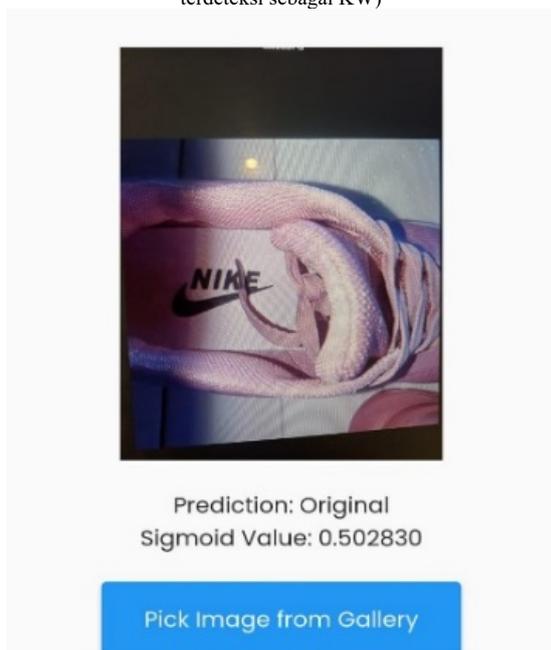
Pengujian di perangkat Android menunjukkan respons cepat, dengan prediksi di bawah satu detik per gambar. Namun, ditemukan perbedaan hasil prediksi saat gambar yang sama diuji dalam konteks berbeda. Gambar KW dari dataset menghasilkan prediksi benar (“KW”, sigmoid 0.495), tetapi saat difoto ulang dengan kamera ponsel, model memprediksi “ORI” (sigmoid 0.502), mengindikasikan sensitivitas model terhadap perubahan kontekstual gambar.



Prediction: KW
Sigmoid Value: 0.495587

Pick Image from Gallery

Gambar 6. Hasil prediksi model terhadap citra asli *dataset* (benar terdeteksi sebagai KW)



Prediction: Original
Sigmoid Value: 0.502830

Pick Image from Gallery

Gambar 7. Hasil prediksi aplikasi terhadap citra KW (diprediksi sebagai ORI)

Gambar 6 dan 7 menunjukkan bahwa model cukup sensitif terhadap perubahan konteks pengambilan gambar, seperti sudut pandang, pencahayaan, dan *noise* kamera. Nilai *sigmoid* yang mendekati ambang batas 0.5 menandakan keraguan model dalam klasifikasi, di mana perubahan kecil pada input dapat mengubah hasil prediksi.

Temuan ini menekankan pentingnya pengujian di dunia nyata, karena meskipun performa kuantitatif tinggi, model tetap memiliki keterbatasan yang perlu dipahami, terutama dalam penerapan *mobile* di luar lingkungan pelatihan.

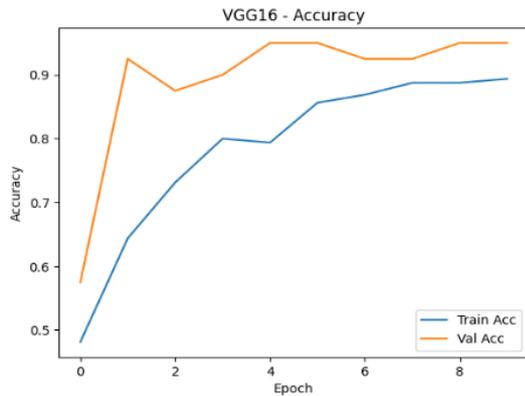
Ambiguitas ini terjadi karena nilai sigmoid berada sangat dekat dengan ambang batas klasifikasi (0.5), yang menunjukkan ketidakpastian tinggi model. Hal ini umumnya dipicu oleh *noise* visual, ketidaksesuaian distribusi data pelatihan dengan kondisi real-world, serta keterbatasan generalisasi model terhadap konteks pencahayaan dan sudut pandang baru. Perbaikan bisa dilakukan melalui peningkatan jumlah data dan *fine-tuning* ulang dengan gambar-gambar dari kondisi nyata.

4.4. Perbandingan *MobileNetV2* dengan VGG16

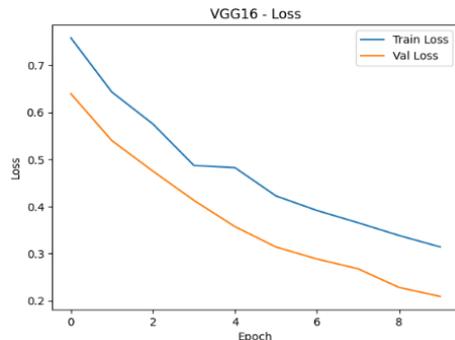
Sebagai bagian dari analisis performa model, penelitian ini membandingkan arsitektur utama yang digunakan, yaitu *MobileNetV2*, dengan salah satu model CNN klasik, yaitu VGG16. Tujuan dari perbandingan ini adalah untuk mengetahui apakah model *lightweight* berbasis *transfer learning* seperti *MobileNetV2* dapat bersaing atau bahkan melampaui model berukuran besar seperti VGG16 dalam hal akurasi klasifikasi, efisiensi waktu, serta kesiapan implementasi di aplikasi *mobile*.

Kedua model dilatih menggunakan pendekatan *transfer learning* dengan *pretrained weights* dari *ImageNet*. Arsitektur *MobileNetV2* dibekukan sepenuhnya (*trainable = False*) dan ditambahkan beberapa lapisan tambahan, seperti *GlobalAveragePooling2D*, *dropout*, dan *dense sigmoid* di bagian output. Sementara itu, VGG16 juga dimanfaatkan sebagai *feature extractor* dengan bobot dibekukan, lalu diikuti dengan *Dense layer* serupa.

Hasil pelatihan model VGG16 menunjukkan peningkatan akurasi yang stabil dari awal hingga akhir *epoch*. Grafik akurasi pelatihan dan validasi ditampilkan pada Gambar 8 dan 9.

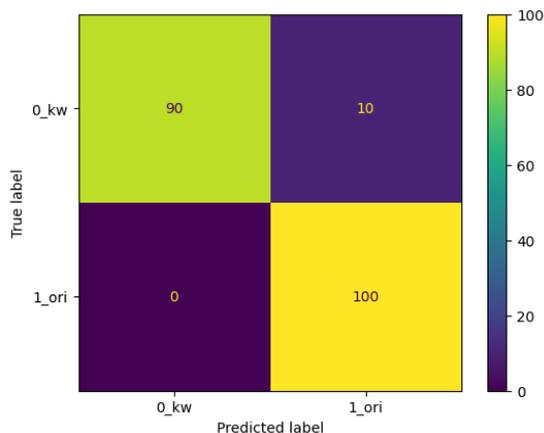


Gambar 8. Grafik akurasi model VGG16



Gambar 9. Grafik loss model VGG16

Akumulasi akurasi validasi VGG16 meningkat tajam dari 57% di *epoch* pertama menjadi 93% pada *epoch* kedua, lalu stabil di kisaran 93–95% hingga akhir pelatihan. Ini menunjukkan kemampuan model dalam belajar cepat meski dengan *dataset* terbatas. Akurasi akhir pada data uji mencapai 95%, sedikit lebih rendah dari *MobileNetV2* yang mencatat akurasi 96% (lihat Bab 4.2). Untuk melihat distribusi klasifikasi yang benar dan salah, dilakukan evaluasi menggunakan *confusion matrix*. *Confusion matrix* untuk model VGG16 ditunjukkan pada Gambar 10 berikut:



Gambar 10. *Confusion matrix* model VGG16 pada data uji

Berdasarkan *confusion matrix*, VGG16 mengklasifikasikan semua gambar ORI dengan

benar (*recall* ORI = 1.00), namun salah mengklasifikasikan 10 gambar KW sebagai ORI (*recall* KW = 0.90). Ini menunjukkan kecenderungan model terlalu percaya diri terhadap kelas ORI, yang berisiko jika diterapkan pada autentikasi produk.

Sebaliknya, *MobileNetV2* menunjukkan hasil lebih seimbang, dengan hanya 6 kesalahan pada KW dan 1 pada ORI. Model ini lebih sensitif terhadap sepatu KW, meskipun sedikit kurang presisi dalam mendeteksi sepatu ORI.

Ringkasan performa kedua model dapat dilihat pada Tabel 3 berikut:

Tabel 3. Perbandingan model *MobileNetV2* dan VGG16

| Model | Ak. Uji | Rec. KW | Rec. ORI | Uk. Model | Wkt. Pred. | Catatan |
|--------------------|---------|---------|----------|-----------|-------------|---------------------|
| <i>MobileNetV2</i> | 96% | 0.94 | 0.99 | ±4.2 MB | < 1 detik | Ringan, cocok di HP |
| VGG16 | 95% | 0.90 | 1.00 | ±528 MB | > 1–2 detik | kurang cocok di HP |

Dari Tabel 3 tersebut dapat disimpulkan bahwa meskipun VGG16 menawarkan performa kuantitatif yang tinggi, *MobileNetV2* tetap menjadi pilihan yang lebih seimbang dalam mendeteksi kedua kelas, dan lebih sesuai untuk diterapkan dalam aplikasi *mobile* yang memerlukan efisiensi, kecepatan, serta sensitivitas yang tinggi terhadap deteksi barang KW.

5. Kesimpulan

Model *MobileNetV2* berhasil mengklasifikasikan sepatu Nike Original dan KW dengan akurasi 96% pada data uji, dan menunjukkan performa lebih seimbang dibanding VGG16. Selain akurat, *MobileNetV2* lebih efisien dalam ukuran dan kecepatan *inferensi*, sehingga lebih cocok untuk perangkat *mobile*. Meskipun VGG16 memiliki *recall* tinggi untuk kelas original, ukurannya yang besar dan kesulitan mengenali sepatu KW menjadi kendala.

Dengan hasil tersebut, *MobileNetV2* dinilai lebih optimal untuk klasifikasi dua kelas dalam skala kecil. Pengembangan ke depan disarankan mencakup perluasan *dataset* dengan variasi latar, sudut, dan pencahayaan, serta penerapan visualisasi seperti Grad-CAM dan pengujian pada data dunia nyata guna meningkatkan generalisasi dan keandalan model.

5.1 Keterbatasan

Penelitian ini memiliki beberapa keterbatasan. Pertama, jumlah *dataset* yang digunakan relatif kecil, hanya 200 gambar, sehingga berisiko menurunkan kemampuan generalisasi model. Kedua, model menunjukkan ambiguitas prediksi pada citra dengan konteks visual berbeda dari data pelatihan, seperti pencahayaan rendah atau sudut ekstrim. Ketiga,

proses evaluasi belum melibatkan data dari pengguna langsung atau crowdsourcing.

5.2 Rencana Penelitian Selanjutnya

Penelitian selanjutnya disarankan untuk mengembangkan dataset yang lebih besar dan lebih bervariasi, baik dari sumber maupun kondisi pengambilan gambar. Selain itu, pendekatan fine-tuning terhadap MobileNetV2 dengan layer tambahan adaptif bisa dilakukan agar model lebih sensitif terhadap noise visual. Penerapan explainable AI (seperti Grad-CAM) juga dapat dijajaki untuk membantu pengguna memahami alasan di balik hasil klasifikasi. Terakhir, integrasi sistem dengan aplikasi mobile sebaiknya diperluas dengan fitur pelaporan dan verifikasi berbasis crowdsourcing.

Daftar Pustaka

- [1] A. Panchal and N. Vora, "Deep Learning based Counterfeit Nike Shoes Detection using YOLOv8 for Object Detection," *J. Innov. Image Process.*, vol. 6, no. 3, pp. 314–328, 2024.
- [2] G. H. Cotte, D. Mellouli, A. Rehman, L. Wang, and D. G. Stork, "Deep neural network-based detection of counterfeit products from smartphone images," 2024.
- [3] R. Bong and C. Lubis, "Pengenalan Nama Model Sepatu Menggunakan Convolutional Neural Network (Cnn) Dengan Model Mobilenet," *J. Ilmu Komput. dan Sist. Inf.*, vol. 12, no. 1, 2024.
- [4] SiBersih, "Cara Mudah untuk Mengetahui Keaslian Sepatu Nike."
- [5] A. N. Hermana, M. G. Husada, and O. Kurniawan, "Penerapan SMOTE Untuk Mengatasi Data Imbalance pada Identifikasi Originalitas Sepatu Converse Menggunakan CNN Arsitektur VGG-16," *J. Pendidik. Tambusai*, vol. 8, no. 1, pp. 10710–10722, 2024.
- [6] A. Z. Lutfan and M. Muhlis, "Jaringan Saraf Tiruan (Neural Network) Untuk Estimasi Produktivitas Pekerja Konstruksi (Artificial Neural Networks for Estimating Construction Worker Productivity)," *J. Daktilitas*, pp. 142–153, 2022.
- [7] A. Raup, W. Ridwan, Y. Khoeriyah, S. Supiana, and Q. Y. Zaqiah, "Deep Learning dan Penerapannya dalam Pembelajaran," *JHIP - J. Ilm. Ilmu Pendidik.*, vol. 5, no. 9, pp. 3258–3267, 2022.
- [8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *Encycl. Mov. Disord. Three-Volume Set*, pp. V2-257-V2-259, 2010.
- [9] G. Annisa and N. Wiliani, "Perbandingan Model CNN dan SVM untuk Klasifikasi Jenis Footwear pada Dataset Alas Kaki Berbasis Citra," vol. 18, no. 1, pp. 28–36, 2025.
- [10] M. L. Alfarisi, M. Herisda, S. A. Kalamasyah, Y. A. Nugraha, S. Salsabila, and M. D. Prasetyo, "Evaluasi Model ANN, CNN, dan SVM Untuk Klasifikasi Gambar CIFAR-10," *J. Anal. Komputasi Digit.*, vol. 9, no. 1, pp. 28–33, 2025.