

# RANCANG BANGUN APLIKASI PEMBELAJARAN INTERAKTIF SISWA BERBASIS ANDROID MENGGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK

Desli Inggrit de Haas<sup>1</sup>, Citra Fathia Palembang<sup>2\*</sup>, Devi Valentino Waas<sup>3</sup>

<sup>1,2,3</sup>Program Studi Ilmu Komputer, Fakultas Sains dan Teknologi, Universitas Pattimura Jl. Ir. M. Putuhena, Ambon, 97233, Indonesia

## Keywords:

Convolutional Neural Network; MobileNetV2; Klasifikasi Citra; Aplikasi Android; Pengenalan Buah.

## Correspondent Email:

[citra.palembang@lecturer.unpatti.ac.id](mailto:citra.palembang@lecturer.unpatti.ac.id)

**Abstrak.** Pengenalan jenis buah merupakan materi penting dalam pembelajaran siswa tingkat dasar, namun metode pembelajaran konvensional masih kurang interaktif dan belum memanfaatkan teknologi secara optimal. Oleh karena itu, penelitian ini bertujuan untuk mengembangkan aplikasi Android pengenalan buah berbasis *Convolutional Neural Network* (CNN) dengan arsitektur MobileNetV2 sebagai media pembelajaran interaktif. Metode penelitian yang digunakan adalah *Research and Development* (R&D), yang meliputi studi literatur, pengumpulan dan pra-pemrosesan citra buah, pengembangan dan evaluasi model CNN, implementasi model ke dalam aplikasi Android, serta pengujian aplikasi. Dataset yang digunakan terdiri dari 20 kelas buah dan dikembangkan menggunakan teknik *transfer learning*, kemudian dikonversi ke format *TensorFlow Lite*. Hasil penelitian menunjukkan bahwa model CNN MobileNetV2 mencapai akurasi pengujian sebesar 90% dengan performa yang baik. Aplikasi Android *Studying Fruits* berhasil diimplementasikan dan seluruh fitur berjalan sesuai fungsinya. Aplikasi ini dapat digunakan sebagai media pembelajaran berbasis kecerdasan buatan yang interaktif dan efektif.



Copyright © [JITET](http://www.jitet.org) (Jurnal Informatika dan Teknik Elektro Terapan). This article is an open access article distributed under terms and conditions of the Creative Commons Attribution (CC BY NC)

**Abstract.** *Fruit recognition is an important learning topic for elementary school students; however, conventional learning methods are often less interactive and do not optimally utilize technology. Therefore, this study aims to develop an Android-based fruit recognition application using a Convolutional Neural Network (CNN) with the MobileNetV2 architecture as an interactive learning medium. This research employs the Research and Development (R&D) method, including literature review, fruit image data collection and preprocessing, CNN model development and evaluation, implementation into an Android application, and application testing. The dataset consists of 20 fruit classes, developed using transfer learning and converted into TensorFlow Lite format. The results show that the MobileNetV2 CNN model achieved a testing accuracy of 90% with good performance. The Studying Fruits Android application functions properly and can be used as an effective and interactive AI-based learning medium.*

## 1. PENDAHULUAN

Perkembangan teknologi informasi telah mengubah cara belajar di bidang pendidikan, terutama dengan adanya perangkat mobile berbasis Android yang bisa digunakan sebagai media belajar yang fleksibel, interaktif, dan mudah diakses. Di Sekolah Dasar (SD), salah satu tantangan utama adalah menyajikan materi pelajaran secara menarik dan interaktif agar bisa meningkatkan semangat belajar serta pemahaman siswa di tengah perkembangan teknologi [1][2]. Untuk menghadapi tantangan ini, diperlukan media pembelajaran yang tidak hanya mudah diakses, tetapi juga bisa memberikan pengalaman belajar yang menarik dan menyenangkan bagi siswa. Salah satu cara efektif untuk meningkatkan minat belajar adalah dengan memilih objek pembelajaran yang dekat dengan kehidupan sehari-hari, seperti pengenalan buah-buahan.

Buah merupakan objek nyata yang sering digunakan sebagai bahan ajar dalam pelajaran di SD, khususnya pada mata pelajaran IPA dan Bahasa Inggris [3][4]. Namun, metode pembelajaran tradisional seperti gambar statis atau penjelasan lisan sering kali kurang menarik dan tidak mampu memberikan pengalaman belajar yang dalam [1][3]. Keterbatasan ini dapat menghambat proses pemahaman dan menurunkan daya ingat siswa [2]. Oleh karena itu, diperlukan media pembelajaran yang menggabungkan teknologi pengenalan gambar dengan interaktivitas agar pemahaman siswa meningkat [3]. Yang dimana Interaktivitas itu merupakan konsep yang merujuk pada derajat partisipasi aktif dan timbal balik antara pengguna (*user*) dengan sistem, media, atau lingkungan. Untuk mengatasi kendala tersebut, diperlukan pendekatan baru yang menggabungkan teknologi pengolahan gambar dengan media interaktif. Salah satu metode yang bisa dipakai untuk mewujudkan hal ini adalah *Convolutional Neural Network* (CNN), yang telah banyak digunakan dalam berbagai bidang pengenalan gambar.

CNN adalah salah satu algoritma *deep learning* yang paling efektif untuk mengolah dan menganalisis data visual, seperti citra atau gambar [5][6]. CNN sangat cocok untuk tugas-tugas klasifikasi citra, di mana model dilatih untuk mengidentifikasi objek dalam sebuah gambar [7][8][9]. Cara kerjanya adalah dengan mengekstraksi fitur-fitur penting dari gambar

melalui lapisan konvolusi, kemudian mengklasifikasikan gambar berdasarkan fitur-fitur tersebut [5].

Metode CNN sudah terbukti efektif dalam mengklasifikasikan gambar, termasuk mengenali jenis buah, dengan tingkat akurasi yang tinggi [5]. CNN mampu mengekstrak fitur secara otomatis dari gambar melalui proses konvolusi dan *pooling*, sehingga lebih akurat dibandingkan metode yang biasa digunakan. Beberapa penelitian sudah memanfaatkan CNN dalam aplikasi berbasis Android, seperti mengenali tokoh wayang [10], mengklasifikasikan tingkat kematangan papaya [11], dan media pembelajaran untuk mengenali buah bagi anak-anak [12]. Keberhasilan CNN dalam berbagai penelitian menunjukkan bahwa metode ini bisa diaplikasikan dalam media pembelajaran berbasis Android, terutama untuk membantu siswa SD dalam mengenali buah.

Media pembelajaran berbasis Android yang interaktif terbukti mampu meningkatkan minat belajar, respons positif siswa, dan hasil belajar mereka [1][4]. Platform Android menawarkan kemudahan, portabilitas, dan kemampuan menampilkan berbagai bentuk konten multimedia. Dengan menggabungkan kemampuan CNN dalam mengenali gambar buah dan media pembelajaran yang interaktif, maka dapat dikembangkan aplikasi yang membantu siswa memahami berbagai jenis buah secara menyenangkan dan bermakna.

Beberapa penelitian sudah mengembangkan media pembelajaran berbasis Android dan aplikasi pengenalan objek dengan CNN, seperti mengklasifikasikan gambar buah [5], mengenali tokoh wayang [10], dan aplikasi *Fruit Zone* berbasis *deep learning* [13]. Penelitian lain juga menegaskan bahwa media interaktif efektif dalam meningkatkan hasil belajar siswa [1][2][3]. Oleh karena itu, sebagian besar studi masih fokus pada aspek pengenalan objek atau media interaktif secara terpisah, tanpa menggabungkan CNN dengan media pembelajaran Android yang khusus ditujukan untuk siswa SD dalam konteks pengenalan buah. Berdasarkan penelitian-penelitian yang telah dilakukan tersebut, maka penulis pada penelitian ini mengembangkan media pembelajaran berbasis Android yang bisa mengklasifikasikan gambar buah dengan CNN sekaligus memberikan pengalaman belajar yang interaktif dan sesuai kebutuhan siswa SD.

## 2. TINJAUAN PUSTAKA

### 2.1. Media Pembelajaran Interaktif

Media pembelajaran interaktif adalah alat yang dibuat untuk membantu proses belajar dengan menggunakan teknologi, sehingga siswa bisa berinteraksi langsung dengan materi yang dipelajari. Media ini tidak hanya memberi informasi dari satu pihak, tetapi juga memungkinkan siswa untuk merespons, memilih, dan mengontrol sendiri jalannya belajar [14]. Dengan demikian, media pembelajaran interaktif bisa menciptakan pengalaman belajar yang lebih bermakna, lebih menarik, dan sesuai dengan kebutuhan siswa.

Menurut peneliti sebelumnya, media pembelajaran interaktif membuat siswa lebih aktif dalam membangun pengetahuan karena mereka tidak hanya menerima informasi, tetapi juga menjadi bagian dari proses belajar [2]. Peneliti lainnya juga menyatakan bahwa penggunaan multimedia interaktif di jenjang sekolah dasar dapat meningkatkan pemahaman konsep secara signifikan, terutama dalam bidang *sains* [3].

### 2.2. Convolutional Neural Network (CNN)

CNN merupakan salah satu jenis *deep learning* yang dirancang khusus untuk melakukan pengolahan data citra digital. CNN terinspirasi dari cara kerja sistem visual manusia dalam mengenali pola melalui proses hierarki fitur [15]. CNN sangat efektif dalam mengenali objek pada gambar karena mampu mengekstraksi fitur penting secara otomatis melalui operasi konvolusi [5]. Selain itu CNN mampu melakukan klasifikasi gambar dengan tingkat akurasi tinggi karena menggunakan pendekatan *feature learning*, di mana fitur tidak ditentukan secara manual melainkan dipelajari secara langsung dari data [7]. Hal ini membuat CNN lebih unggul dibandingkan metode klasifikasi tradisional. Berikut ini ada beberapa persamaan yang digunakan:

$$f(x)_{ReLU} = \max(0, x) \quad (1)$$

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \quad (2)$$

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives} \quad (3)$$

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

$$Accuracy = \frac{Jumlah\ Prediksi\ Benar}{Jumlah\ Total\ Prediksi} \quad (5)$$

### 2.3. Arsitektur MobileNetV2

MobileNetV2 merupakan salah satu arsitektur CNN yang dirancang untuk kebutuhan komputasi ringan, khususnya pada perangkat dengan sumber daya terbatas seperti perangkat mobile. Arsitektur ini dikembangkan sebagai penyempurnaan dari MobileNet generasi pertama dengan tujuan meningkatkan efisiensi komputasi dan akurasi model tanpa menambah kompleksitas jaringan secara signifikan [16]. Ciri utama MobileNetV2 terletak pada penggunaan *inverted residual block* dan *linear bottleneck*. Berbeda dengan CNN konvensional yang memperkecil dimensi fitur pada awal jaringan, MobileNetV2 justru memperluas dimensi fitur terlebih dahulu, kemudian melakukan operasi konvolusi secara terpisah (*depthwise separable convolution*), dan akhirnya mengompresi kembali fitur tersebut. Pendekatan ini terbukti mampu mengurangi jumlah parameter dan operasi komputasi secara signifikan [16][17].

Selain itu, MobileNetV2 menggunakan *linear activation function* pada lapisan *bottleneck* untuk mencegah hilangnya informasi penting akibat fungsi aktivasi *non-linear* seperti ReLU menggunakan persamaan (1). Strategi ini memungkinkan model mempertahankan representasi fitur yang lebih baik, terutama pada lapisan dengan dimensi rendah. Hasil analisis performa menunjukkan bahwa MobileNetV2 memiliki keseimbangan yang baik antara akurasi dan efisiensi dibandingkan arsitektur CNN lainnya [17].

Dalam implementasinya, MobileNetV2 banyak digunakan bersama teknik *transfer learning*, yaitu memanfaatkan bobot awal dari model yang telah dilatih pada dataset besar seperti *ImageNet*. Arsitektur yang dapat ditransfer memiliki kemampuan generalisasi yang baik pada berbagai tugas klasifikasi Citra [18]. Hal ini diperkuat oleh penelitian lainnya yang membuktikan bahwa MobileNetV2 dengan *transfer learning* mampu menghasilkan akurasi tinggi pada klasifikasi Citra buah-buahan [19].

Selanjutnya, penelitian lain juga menunjukkan bahwa MobileNetV2 memiliki performa akurasi yang baik dalam klasifikasi

Citra medis *X-ray*, meskipun dijalankan pada arsitektur yang relatif ringan [20]. Temuan ini menegaskan bahwa MobileNetV2 tidak hanya efisien secara komputasi, tetapi juga andal dalam mengekstraksi fitur visual dari berbagai jenis citra. Berdasarkan karakteristik tersebut, MobileNetV2 dipilih dalam penelitian ini karena memiliki keunggulan dalam hal efisiensi, kecepatan inferensi, dan kemampuan generalisasi yang baik [21]. Arsitektur ini sangat sesuai untuk diimplementasikan pada aplikasi pengenalan buah berbasis Android sebagai media pembelajaran interaktif bagi siswa SD, yang menuntut performa cepat, ukuran model kecil, dan akurasi yang memadai.

#### 2.4. Metode Pengujian Aplikasi

Pengujian aplikasi adalah bagian penting dalam pengembangan perangkat lunak agar memastikan sistem berjalan sesuai dengan apa yang dibutuhkan, tidak ada kesalahan, dan memberikan pengalaman yang menyenangkan bagi pengguna. Pengujian perangkat lunak adalah proses yang terstruktur untuk menilai kualitas, keandalan, dan kemampuan bekerja sistem tersebut. Dalam penelitian ini, pengujian aplikasi dilakukan bukan hanya untuk memastikan bahwa metode CNN mampu mengklasifikasikan buah secara akurat, tetapi juga untuk mengecek apakah aplikasi ini layak digunakan sebagai media pembelajaran interaktif berbasis Android di SD. Beberapa metode pengujian yang cocok digunakan adalah Pengujian Fungsional (*Black Box Testing*), Pengujian Kinerja (*Performance Testing*), dan Pengujian Akurasi Model CNN. *Black box testing* digunakan untuk menguji cara kerja aplikasi tanpa perlu mengetahui bagian-bagian dalam program tersebut. Pengujian kinerja bertujuan untuk mengetahui seberapa cepat, stabil, dan efisien aplikasi ketika digunakan. Pengujian akurasi dilakukan agar mengetahui seberapa baik CNN mampu mengenali dan mengklasifikasikan buah.

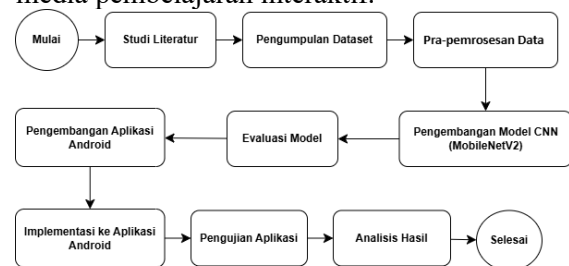
#### 2.5. Analisis Data

Analisis data adalah proses yang sistematis untuk memproses, memahami, serta mengambil kesimpulan dari data yang didapat agar bisa menjawab permasalahan dalam penelitian. Dalam pengembangan aplikasi pengenalan buah berbasis Android menggunakan metode CNN, analisis data dilakukan pada tahap

pemrosesan gambar serta tahap evaluasi kinerja aplikasi sebagai alat pembelajaran interaktif. Analisis data adalah cara untuk mencari dan mengatur data yang diperoleh dari hasil wawancara, catatan lapangan, atau dokumen secara sistematis agar mudah dipahami dan bisa disampaikan kepada orang lain. Dalam penelitian berbasis aplikasi, tujuan analisis data adalah mengukur kinerja sistem serta mengetahui tanggapan pengguna terhadap aplikasi yang dikembangkan.

### 3. METODE PENELITIAN

Metode penelitian yang digunakan adalah penelitian dan pengembangan (*Research and Development*). Model ini dipilih karena penelitian ini tidak hanya ingin menguji algoritma CNN dalam mengenali gambar buah, tetapi juga mengembangkan sebuah produk berupa aplikasi Android yang bisa digunakan sebagai sarana pembelajaran interaktif bagi siswa SD. Produk yang diteliti adalah aplikasi pengenalan buah berbasis Android menggunakan CNN yang berfungsi sebagai media pembelajaran interaktif.



Gambar 1. Flowchart Tahapan Penelitian

Berikut ini adalah penjelasan dari *Flowchart* Tahapan Penelitian di atas:

#### 3.1. Studi Literatur

Melakukan pemahaman tentang teori, jurnal, dan penelitian sebelumnya yang berkaitan dengan media pembelajaran interaktif berbasis Android, penerapan CNN dalam pengklasifikasian gambar buah serta penelitian serupa yang menggabungkan *deep learning* dalam aplikasi Android.

#### 3.2. Pengumpulan Dataset

Dataset buah diperoleh dari sumber terbuka seperti *Kaggle*. Berikut ini adalah link *Kaggle*: <https://www.kaggle.com/datasets/marquis03/fruits-100>. Buah yang dipilih antara lain alpukat, anggur, apel, durian, jambu, jambu biji, kelapa,

langsap, lemon, mangga, manggis, nanas, nangka, pepaya, pisang, rambutan, salak, semangka, sirsak, stroberi. Selanjutnya, data dilabeli agar bisa digunakan dalam model CNN.

### 3.3. *Pra-pemrosesan Data (Preprocessing)*

Pra-pemrosesan Data (*Preprocessing*) Dataset gambar buah diperlakukan dengan mengubah ukuran gambar menjadi  $224 \times 224$  piksel dan normalisasi pixel dengan nilai 0 sampai 1. Data *augmentation* seperti rotasi, *flip*, dan *zooming* dilakukan untuk meningkatkan variasi dataset.

### 3.4. *Pengembangan Model CNN (MobileNetV2)*

Pada tahap ini dilakukan perancangan dan pelatihan model CNN menggunakan arsitektur MobileNetV2. Model dikembangkan dengan memanfaatkan teknik *transfer learning* dan disesuaikan untuk klasifikasi citra buah sesuai dengan jumlah kelas yang ditentukan.

### 3.5. *Evaluasi Model*

Model yang telah dilatih kemudian dievaluasi menggunakan data uji. Evaluasi dilakukan dengan menggunakan metrik kinerja seperti akurasi, *precision*, *recall*, dan *F1-score* untuk mengetahui tingkat keberhasilan model dalam melakukan klasifikasi citra buah.

### 3.6. *Pengembangan Aplikasi Android*

Tahap ini berfokus pada perancangan dan pembuatan antarmuka serta fitur aplikasi Android, seperti halaman *login*, halaman utama, fitur klasifikasi buah, kuis pembelajaran, dan profil pengguna. Pengembangan dilakukan menggunakan Android Studio dengan bahasa pemrograman Kotlin, sebelum model CNN diintegrasikan ke dalam aplikasi. Tahap ini memisahkan antara pembuatan aplikasi secara fungsional dan tampilan dan integrasi kecerdasan buatan ke dalam aplikasi, Sehingga alur penelitian menjadi lebih jelas dan sistematis.

### 3.7. *Implementasi ke Aplikasi Android*

Model CNN yang telah dilatih diekspor ke format *TensorFlow Lite* dan dimitigasikan ke dalam aplikasi Android menggunakan Android Studio. Fitur aplikasi mencakup pengenalan buah melalui kamera atau galeri, penyajian informasi dalam bentuk teks dan audio, serta

kuis interaktif untuk membantu proses belajar siswa.

### 3.8. *Pengujian Aplikasi*

Tahap pengujian aplikasi bertujuan untuk memastikan seluruh fitur aplikasi berjalan dengan baik dan sesuai dengan fungsinya. Pengujian dilakukan untuk menilai kinerja sistem, kestabilan aplikasi, serta kesesuaian hasil klasifikasi yang ditampilkan.

### 3.9. *Analisis Hasil*

Pada tahap ini dilakukan analisis terhadap hasil evaluasi model dan pengujian aplikasi. Analisis bertujuan untuk mengetahui kelebihan, keterbatasan, serta tingkat keberhasilan sistem yang dikembangkan dalam mencapai tujuan penelitian.

## 4. HASIL DAN PEMBAHASAN

### 4.1. *Hasil Pengembangan Model*

#### 4.1.1. *Arsitektur Model CNN*

Model CNN yang dikembangkan pada penelitian ini menggunakan arsitektur MobileNetV2 sebagai *feature extractor*. Model menerima citra masukan berukuran  $224 \times 224$  piksel dengan 3 kanal warna (RGB). MobileNetV2 dipilih karena memiliki jumlah parameter yang relatif kecil namun tetap mampu menghasilkan performa klasifikasi yang baik, sehingga sesuai untuk diimplementasikan pada perangkat bergerak berbasis Android. Berdasarkan hasil *model.summary()*, total parameter pada model adalah 2.424.532 parameter, dengan 166.548 parameter *trainable* dan 2.257.984 parameter *non-trainable*. Parameter *non-trainable* berasal dari bobot MobileNetV2 pra-latih (*pre-trained weights*), sedangkan parameter *trainable* berasal dari lapisan klasifikasi tambahan yang dirancang khusus untuk klasifikasi buah.

Pada bagian akhir model, ditambahkan beberapa lapisan yaitu *Global Average Pooling*, *Dropout*, dan *Dense layer*. *Global Average Pooling* berfungsi mereduksi dimensi fitur, *Dropout* digunakan untuk mengurangi risiko *overfitting*, dan *Dense layer* terakhir dengan fungsi aktivasi *softmax* menghasilkan keluaran berupa probabilitas dari 20 kelas buah. Arsitektur lengkap model CNN berbasis MobileNetV2 yang digunakan dalam penelitian

ini ditampilkan pada Gambar 2, yang memperlihatkan struktur model secara menyeluruh, mulai dari lapisan input citra berukuran  $224 \times 224$  piksel, rangkaian *convolution* dan *depthwise convolution* pada MobileNetV2 sebagai *feature extractor*, hingga lapisan *Global Average Pooling*, *Dropout*, dan *Dense layer* sebagai *classifier*.

Model: "functional"

Layer (type)	Output Shape	Param #	Connected to
input_layer (InputLayer)	(None, 224, 224, 3)	0	-
conv1 (Conv2D)	(None, 112, 112, 32)	864	input_layer[0][0]
bn_conv1 (BatchNormalization)	(None, 112, 112, 32)	128	conv1[0][0]
conv1_relu (ReLU)	(None, 112, 112, 32)	0	bn_conv1[0][0]
expanded_conv_dept... (DepthwiseConv2D)	(None, 112, 112, 32)	288	conv1_relu[0][0]
expanded_conv_dept... (BatchNormalization)	(None, 112, 112, 32)	128	expanded_conv_dept...
expanded_conv_dept... (ReLU)	(None, 112, 112, 32)	0	expanded_conv_dept...
expanded_conv_proj... (Conv2D)	(None, 112, 112, 16)	512	expanded_conv_dept...
expanded_conv_proj... (BatchNormalization)	(None, 112, 112, 16)	64	expanded_conv_proj...
block_1_expand (Conv2D)	(None, 112, 112, 96)	1,536	expanded_conv_proj...
block_1_expand_BN (BatchNormalization)	(None, 112, 112, 96)	384	block_1_expand[0...]
block_1_expand_relu (ReLU)	(None, 112, 112, 96)	0	block_1_expand_BN...
block_1_pad (ZeroPadding2D)	(None, 113, 113, 96)	0	block_1_expand_re...
conv_1 (Conv2D)	(None, 7, 7, 1280)	409,600	block_16_project...
conv_1_bn (BatchNormalization)	(None, 7, 7, 1280)	5,120	conv_1[0][0]
out_relu (ReLU)	(None, 7, 7, 1280)	0	conv_1_bn[0][0]
global_average_pool... (GlobalAveragePool...)	(None, 1280)	0	out_relu[0][0]
dropout (Dropout)	(None, 1280)	0	global_average_p...
dense (Dense)	(None, 128)	163,968	dropout[0][0]
dropout_1 (Dropout)	(None, 128)	0	dense[0][0]
dense_1 (Dense)	(None, 20)	2,580	dropout_1[0][0]

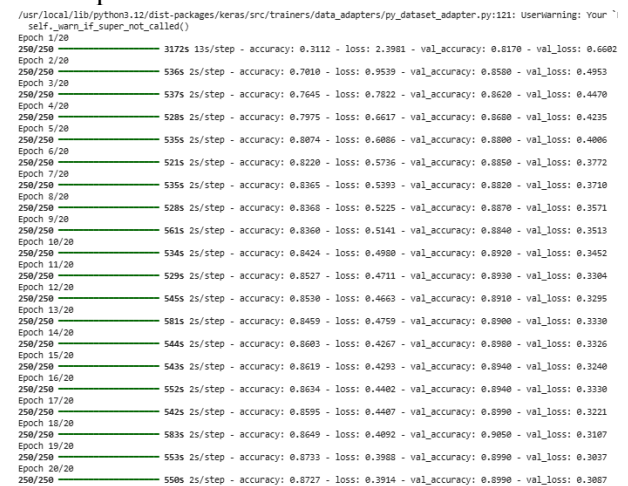
Total params: 2,424,532 (9.25 MB)  
 Trainable params: 166,548 (650.58 KB)  
 Non-trainable params: 2,257,984 (8.61 MB)

Gambar 2. Arsitektur MobileNetV2

#### 4.1.2. Proses Pelatihan Model

Proses pelatihan model dilakukan selama 20 *epoch* dengan jumlah 250 *steps* per *epoch*, sebagaimana ditunjukkan pada hasil pelatihan model. Berdasarkan proses tersebut, terlihat adanya peningkatan performa model yang konsisten dari *epoch* awal hingga *epoch* akhir. Pada *epoch* pertama, nilai akurasi pelatihan masih relatif rendah, yaitu sebesar 31,12%, dengan nilai *loss* sebesar 2,3981, sementara akurasi validasi telah mencapai 81,70% dengan *validation loss* sebesar 0,6602. Seiring bertambahnya *epoch*, model menunjukkan kemampuan belajar yang semakin baik, ditandai dengan meningkatnya nilai akurasi dan menurunnya nilai *loss* baik pada data latih maupun data validasi.

Pada *epoch* ke-20, model mencapai akurasi pelatihan sebesar 87,27% dengan *loss* sebesar 0,3914. Sementara itu, akurasi validasi mencapai 89,90% dengan *validation loss* sebesar 0,3087. Hasil ini menunjukkan bahwa model tidak hanya mampu mempelajari pola dari data latih secara efektif, tetapi juga memiliki kemampuan generalisasi yang baik terhadap data validasi, serta tidak menunjukkan indikasi *overfitting* yang signifikan. Perkembangan performa model CNN selama proses pelatihan, yang ditunjukkan melalui perubahan nilai *accuracy* yang dihitung menggunakan persamaan (5) dan *loss* pada data latih dan data validasi di setiap *epoch*, dapat dilihat pada Gambar 3.



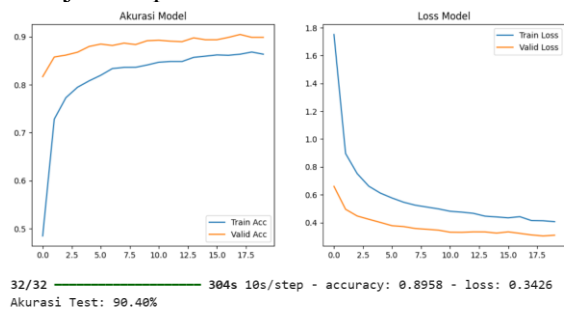
Gambar 3. Pelatihan Model

#### 4.1.3. Hasil Evaluasi Model Pada Data Uji

Evaluasi model dilakukan menggunakan data uji yang terdiri dari 1.000 citra, dengan masing-masing kelas buah berjumlah 50 citra. Berdasarkan hasil pengujian, model CNN memperoleh akurasi data uji sebesar 90,40% dengan nilai *loss* sebesar 0,3426. Hasil ini menunjukkan bahwa model memiliki kemampuan yang baik dalam mengklasifikasikan citra buah yang belum pernah dilihat sebelumnya. Berdasarkan grafik akurasi dan *loss* selama proses pelatihan dan validasi, terlihat bahwa nilai akurasi data latih dan data validasi mengalami peningkatan yang stabil hingga akhir *epoch*, sementara nilai *loss* menunjukkan tren penurunan yang konsisten. Pola ini mengindikasikan bahwa model mampu mempelajari fitur secara efektif tanpa mengalami *overfitting* yang signifikan, karena kurva data latih dan data validasi cenderung

berdekatan. Selain akurasi, evaluasi performa model juga dilakukan menggunakan metrik *precision*, *recall*, dan *F1-score* menggunakan persamaan (2), (3), dan (4).

Hasil evaluasi menunjukkan bahwa nilai *macro average precision*, *recall*, dan *F1-score* berada pada kisaran 0,90, yang menandakan bahwa performa model relatif seimbang pada seluruh kelas buah. Beberapa kelas buah seperti Nanas, Pisang, Rambutan, dan Salak menunjukkan performa yang sangat baik dengan nilai *precision* dan *recall* mendekati atau mencapai 1,00. Namun demikian, terdapat beberapa kelas dengan performa yang relatif lebih rendah, seperti Mangga, yang memiliki nilai *precision* 0,67 dan *recall* 0,76. Hal ini diduga disebabkan oleh kemiripan visual antara buah mangga dengan beberapa kelas buah lain, sehingga meningkatkan kemungkinan terjadinya kesalahan klasifikasi. Grafik perbandingan nilai akurasi dan *loss* pada data latih serta data validasi selama proses pelatihan, serta hasil evaluasi model pada data uji, ditunjukkan pada Gambar 4.

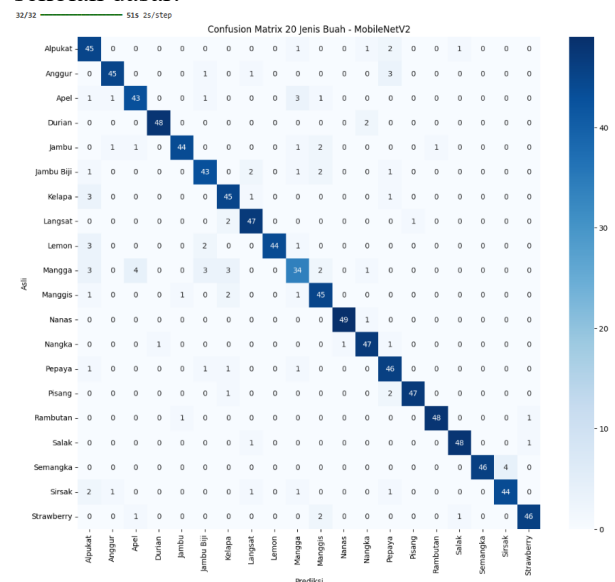


Gambar 4. Hasil Evaluasi Model

#### 4.1.4. Analisis Hasil Klasifikasi

Secara keseluruhan, hasil pengembangan model CNN berbasis MobileNetV2 menunjukkan performa yang baik dalam tugas klasifikasi 20 jenis buah. Hal ini ditunjukkan oleh *confusion matrix* pada Gambar 5, di mana sebagian besar citra uji terklasifikasi dengan benar dan nilai prediksi dominan berada pada diagonal utama, yang menandakan tingkat kesalahan klasifikasi yang relatif rendah. Selain itu, berdasarkan hasil evaluasi yang disajikan pada Tabel 1, model mencapai akurasi sebesar 90%, dengan nilai *macro average precision*, *recall*, dan *F1-score* masing-masing sebesar 0,91; 0,90; dan 0,90. Nilai-nilai tersebut menunjukkan bahwa performa model relatif seimbang pada seluruh kelas buah.

Beberapa kelas seperti Nanas, Pisang, Rambutan, dan Salak memiliki performa yang sangat baik dengan nilai *precision* dan *recall* mendekati atau mencapai 1,00, sementara kelas Mangga menunjukkan performa yang lebih rendah akibat adanya kemiripan visual dengan beberapa jenis buah lainnya. Dengan mempertimbangkan hasil evaluasi tersebut, dapat disimpulkan bahwa penggunaan arsitektur MobileNetV2 yang dikombinasikan dengan lapisan klasifikasi tambahan mampu menghasilkan model yang akurat, efisien, dan stabil. Model yang dikembangkan telah memenuhi tujuan penelitian, yaitu menghasilkan sistem klasifikasi buah yang dapat diimplementasikan secara efektif pada perangkat bergerak berbasis Android sebagai media pembelajaran interaktif bagi siswa sekolah dasar.



Gambar 4. Confusion Matrix

Tabel 1. Precision, Recall, dan F1-score

	Precision	Recall	F1-score	Support
Alpukat	0.75	0.90	0.82	50
Anggur	0.94	0.90	0.92	50
Apel	0.88	0.86	0.87	50
Durian	0.98	0.96	0.97	50
Jambu	0.96	0.88	0.92	50
Jambu Biji	0.84	0.86	0.85	50
Kelapa	0.83	0.90	0.87	50
Langsat	0.89	0.94	0.91	50
Lemon	1.00	0.88	0.94	50
Mangga	0.77	0.68	0.72	50
Manggis	0.83	0.90	0.87	50
Nanas	0.98	0.98	0.98	50
Nangka	0.90	0.94	0.92	50
Pepaya	0.81	0.92	0.86	50

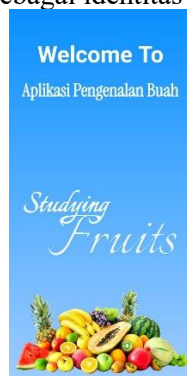
Pisang	0.98	0.94	0.96	50
Rambutan	0.98	0.96	0.97	50
Salak	0.96	0.96	0.96	50
Semangka	1.00	0.92	0.96	50
Sirsak	0.92	0.88	0.90	50
Strawberry	0.96	0.92	0.94	50
Accuracy			0.90	1000
Macro avg	0.91	0.90	0.90	1000
Weighted avg	0.91	0.90	0.90	1000

## 4.2. Hasil Implementasi Aplikasi Android

### 4.2.1. Deskripsi Umum Aplikasi

Aplikasi Android yang dikembangkan pada penelitian ini merupakan aplikasi media pembelajaran interaktif pengenalan buah yang memanfaatkan model CNN hasil pelatihan sebelumnya. Aplikasi ini dibangun menggunakan Android Studio dengan bahasa pemrograman Kotlin, serta mengintegrasikan model klasifikasi buah dalam format *TensorFlow Lite (TFLite)* sehingga dapat dijalankan secara *offline* pada perangkat Android. Berdasarkan hasil implementasi dan kompilasi aplikasi, aplikasi ini berhasil dijalankan dengan baik pada perangkat Android.

Aplikasi memungkinkan pengguna, khususnya siswa sekolah dasar, untuk melakukan proses klasifikasi buah melalui kamera secara langsung maupun dengan memilih gambar dari galeri, sehingga mendukung proses pembelajaran yang fleksibel dan interaktif. Gambar 5 menunjukkan tampilan *splash screen* aplikasi yang muncul saat aplikasi pertama kali dijalankan. Tampilan ini menampilkan judul *Studying Fruits* dan ilustrasi buah sebagai identitas visual aplikasi.



Gambar 5. Halaman *Splash*

### 4.2.2. Implementasi Model CNN pada Aplikasi

Model CNN berbasis MobileNetV2 yang telah dilatih pada tahap sebelumnya dikonversi ke dalam format *.tflite* dan diintegrasikan ke dalam aplikasi Android. Proses inferensi dilakukan secara *on-device* dengan menyesuaikan ukuran input Citra menjadi  $224 \times 224$  piksel sesuai dengan kebutuhan model. Pada saat pengguna memilih atau mengambil gambar, aplikasi akan melakukan tahap pra-pemrosesan Citra, kemudian menjalankan proses prediksi menggunakan *TensorFlow Lite Interpreter*. Hasil pengujian menunjukkan bahwa model dapat memproses Citra dengan waktu yang relatif singkat dan menampilkan hasil prediksi berupa nama buah serta tingkat kepercayaan (*confidence score*).

Hal ini menunjukkan bahwa model CNN yang dikembangkan kompatibel, ringan, dan berjalan stabil pada lingkungan Android. Gambar 6 menunjukkan tampilan halaman *Home/Klasifikasi Buah*, di mana pengguna dapat memasukkan gambar buah melalui tombol Buka Galeri atau Buka Kamera. Setelah gambar diproses, aplikasi menampilkan hasil klasifikasi berupa nama buah serta menyediakan fitur audio pelafalan (*Text-to-Speech*) dalam Bahasa Inggris.



Gambar 6. Halaman *Home*

### 4.2.3. Implementasi Fitur Kuis Interaktif

Aplikasi dilengkapi dengan fitur kuis interaktif untuk mengukur pemahaman pengguna terhadap materi pengenalan buah. Gambar 7 memperlihatkan tampilan halaman *Quiz Buah*, yang berisi soal pilihan ganda.

Setiap jawaban yang benar akan menambah skor pengguna, dan progres kuis ditampilkan secara real-time.



**Gambar 7.** Halaman *Quiz*

#### 4.2.4. Implementasi Profil Pengguna

Aplikasi menyediakan halaman profil untuk menampilkan biodata pengguna dan hasil belajar. Gambar 8 menunjukkan tampilan Profil Pengguna, yang memuat informasi seperti nama, jenis kelamin, sekolah, kelas, serta skor kuis. Fitur ini memungkinkan pemantauan perkembangan belajar pengguna.

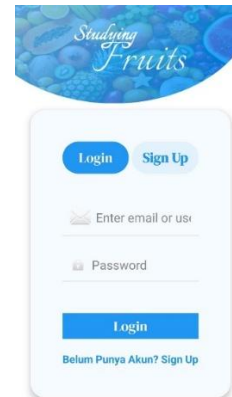


**Gambar 8.** Halaman *Profile*

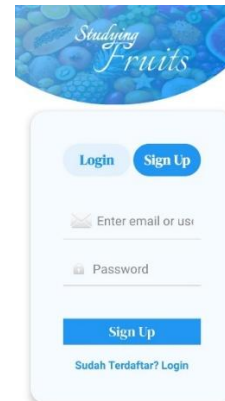
#### 4.2.5. Implementasi Login dan Sign Up

Aplikasi memiliki sistem autentikasi pengguna berupa fitur login dan pendaftaran akun. Gambar 9 menunjukkan tampilan *Login*, yang digunakan pengguna untuk mendaftarkan akun baru. Gambar 10 menunjukkan tampilan *Sign Up*, yang digunakan pengguna untuk masuk ke dalam aplikasi menggunakan akun

yang telah terdaftar. Keberadaan fitur autentikasi ini bertujuan untuk menyimpan data pengguna dan skor kuis secara konsisten.



**Gambar 9.** Halaman *Login*



**Gambar 10.** Halaman *Sign Up*

#### 4.2.6. Pengambilan Gambar Buah

Pada Gambar 11 dan Gambar 12 yang ditampilkan menunjukkan dua metode input citra yang disediakan pada aplikasi Android *Studying Fruits*, yaitu melalui kamera dan galeri. Pada pengambilan gambar melalui kamera yang ditampilkan pada Gambar 11, pengguna dapat langsung memotret objek buah menggunakan kamera *smartphone*. Setelah gambar diambil, pengguna diberikan opsi untuk mengulangi pengambilan gambar (*Coba lagi*) atau melanjutkan proses (*OK*). Gambar yang dipilih kemudian digunakan sebagai input untuk proses klasifikasi oleh model CNN. Sementara itu, pada pengambilan gambar melalui galeri yang ditampilkan pada Gambar 12, pengguna dapat memilih citra buah yang telah tersimpan sebelumnya di perangkat.

Aplikasi menampilkan daftar gambar dalam bentuk grid sehingga memudahkan pengguna dalam memilih citra buah yang diinginkan. Setelah gambar dipilih, citra tersebut akan diproses dan diklasifikasikan oleh model CNN MobileNetV2.

Kedua metode input ini memberikan fleksibilitas kepada pengguna dalam menggunakan aplikasi, baik untuk pengujian secara langsung menggunakan kamera maupun menggunakan dataset atau gambar yang telah tersedia di galeri perangkat. Hasil klasifikasi yang ditampilkan menunjukkan bahwa aplikasi mampu memproses citra dari kedua sumber dengan baik. Namun, terkadang ada beberapa hasil klasifikasi citra yang kurang baik karena gambar yang dimasukkan tidak sesuai dengan dataset citra buah yang ada. Karena itu jika gambar buah yang dimasukkan tidak terdapat pada dataset maka akan muncul hasil klasifikasi buah tidak dikenali.



**Gambar 11.** Tampilan Pengambilan Gambar Melalui Kamera



**Gambar 12.** Tampilan Pengambilan Gambar Melalui Galeri

#### 4.2.7. Analisis Hasil Implementasi

Berdasarkan hasil implementasi dan pengujian aplikasi Android, dapat disimpulkan bahwa integrasi model CNN MobileNetV2 ke dalam aplikasi berjalan dengan baik. Aplikasi mampu memberikan hasil klasifikasi buah secara akurat dan responsif, sehingga dapat digunakan sebagai media pembelajaran berbasis teknologi untuk membantu siswa mengenal berbagai jenis buah. Hasil ini menunjukkan bahwa aplikasi Android yang dikembangkan telah memenuhi tujuan penelitian, yaitu menghasilkan sistem pembelajaran interaktif yang menggabungkan kecerdasan buatan dengan perangkat *mobile* secara efektif.

#### 4.3. Hasil Pengujian Aplikasi

Pengujian aplikasi dilakukan untuk mengetahui tingkat keberfungsian, kinerja, dan keandalan aplikasi *Studying Fruits* setelah diimplementasikan pada perangkat Android. Pengujian meliputi pengujian fungsional, pengujian kinerja sistem, serta pengujian akurasi hasil klasifikasi pada aplikasi.

##### 4.3.1. Pengujian Fungsional (Black Box Testing)

Pengujian fungsional dilakukan menggunakan metode *Black Box Testing*, yaitu dengan menguji setiap fitur aplikasi tanpa memperhatikan struktur kode program. Pengujian difokuskan pada kesesuaian keluaran aplikasi terhadap fungsi yang diharapkan. Hasil pengujian menunjukkan bahwa seluruh fitur utama aplikasi dapat berjalan dengan baik. Proses *login* dan *sign up* pengguna berjalan sesuai dengan fungsinya. Halaman *home* dapat ditampilkan dengan baik dan navigasi menu berfungsi normal. Fitur pengambilan gambar melalui kamera dan pemilihan gambar dari galeri berjalan dengan baik. Proses klasifikasi buah menggunakan model CNN dapat menampilkan nama buah secara tepat. Fitur kuis interaktif dapat menampilkan soal, memproses jawaban, dan menghitung skor. Halaman profil pengguna dapat menampilkan biodata dan skor pengguna. Berdasarkan hasil tersebut, dapat disimpulkan bahwa secara fungsional aplikasi telah berjalan sesuai dengan kebutuhan sistem.

#### 4.3.2. Pengujian Kinerja Aplikasi

Pengujian kinerja dilakukan untuk mengetahui kemampuan aplikasi dalam memproses citra dan menampilkan hasil klasifikasi. Pengujian dilakukan dengan menjalankan aplikasi pada perangkat Android dan mengamati waktu respons saat melakukan klasifikasi buah. Hasil pengujian menunjukkan bahwa aplikasi mampu memproses citra buah dan menampilkan hasil prediksi dalam waktu yang relatif singkat. Proses inferensi menggunakan *TensorFlow Lite* berjalan secara stabil tanpa menyebabkan keterlambatan yang signifikan ataupun gangguan pada antarmuka pengguna. Selain itu, penggunaan memori aplikasi masih dalam batas wajar untuk perangkat Android.

#### 4.3.3. Pengujian Akurasi Sistem pada Aplikasi

Pengujian akurasi sistem dilakukan dengan membandingkan hasil prediksi aplikasi dengan label sebenarnya dari citra uji. Hasil pengujian menunjukkan bahwa tingkat akurasi klasifikasi pada aplikasi Android sejalan dengan hasil pengujian model CNN pada tahap sebelumnya, dengan tingkat akurasi mendekati 90%. Perbedaan kecil pada hasil prediksi umumnya disebabkan oleh kualitas citra yang diambil melalui kamera, seperti pencahayaan dan sudut pengambilan gambar. Namun secara keseluruhan, sistem mampu mengenali jenis buah dengan tingkat keandalan yang tinggi.

#### 4.3.4. Analisis Hasil Pengujian Aplikasi

Berdasarkan hasil pengujian, aplikasi *Studying Fruits* telah berjalan dengan baik dari sisi fungsionalitas, kinerja, dan akurasi. Seluruh fitur utama aplikasi dapat digunakan sesuai dengan perancangan tanpa mengalami kendala berarti. Proses klasifikasi buah berlangsung cepat dan responsif karena menggunakan model CNN MobileNetV2 yang diimplementasikan dengan *TensorFlow Lite*. Tingkat akurasi klasifikasi pada aplikasi mencapai sekitar 90%, sejalan dengan hasil pengujian model CNN. Perbedaan kecil pada hasil klasifikasi umumnya disebabkan oleh faktor pencahayaan dan kualitas gambar. Secara keseluruhan, aplikasi dinilai stabil, efisien, dan layak digunakan sebagai media pembelajaran interaktif berbasis Android untuk siswa sekolah dasar.

## 5. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa:

- Model CNN dengan arsitektur MobileNetV2 berhasil dikembangkan untuk melakukan klasifikasi Citra buah. Model mampu mengenali 20 jenis buah dengan tingkat akurasi pengujian sebesar 90%, sehingga menunjukkan kinerja yang baik dan stabil.
- Penerapan teknik *transfer learning* pada MobileNetV2 terbukti efektif dalam meningkatkan akurasi model dengan jumlah parameter yang relatif kecil, sehingga cocok digunakan pada perangkat *mobile*.
- Model CNN yang telah dilatih berhasil diimplementasikan ke dalam aplikasi Android *Studying Fruits* menggunakan *TensorFlow Lite*. Aplikasi mampu melakukan klasifikasi buah melalui kamera maupun galeri dengan waktu respons yang cepat.
- Hasil pengujian aplikasi menunjukkan bahwa seluruh fitur utama, seperti pengenalan buah, kuis pembelajaran, dan pengelolaan profil pengguna, dapat berjalan dengan baik sesuai dengan fungsinya.
- Aplikasi *Studying Fruits* dapat dimanfaatkan sebagai media pembelajaran interaktif yang membantu pengguna dalam mengenal berbagai jenis buah dengan memanfaatkan teknologi kecerdasan buatan.

## UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada pihak-pihak terkait yang telah memberi dukungan terhadap penelitian ini.

## DAFTAR PUSTAKA

- [1] K. S. Kartini dan I. N. T. A. Putra, "Pengaruh Penggunaan Media Pembelajaran Interaktif Berbasis Android Terhadap Hasil Belajar Siswa," *J. Redoks J. Pendidik. Kim. Dan Ilmu Kim.*, vol. 3, no. 2, hal. 8–12, 2020, doi: 10.33627/re.v3i2.417.
- [2] U. B. Harsiwi dan L. D. D. Arini, "Pengaruh Pembelajaran Menggunakan Media Pembelajaran Interaktif terhadap Hasil Belajar Siswa di Sekolah Dasar," *J. Basicedu*, vol. 5, no. 5, hal. 3(2), 524–532, 2020, [Daring]. Tersedia pada: <https://journal.uii.ac.id/ajie/article/view/971>

- [3] N. Deliany, A. Hidayat, dan Y. Nurhayati, "Penerapan Multimedia Interaktif untuk Meningkatkan Pemahaman Konsep IPA Peserta Didik di Sekolah Dasar," *Educ. J. Pendidik. Dan Pembelajaran*, vol. 17, no. 2, hal. 90–97, 2019, [Daring]. Tersedia pada: <https://doi.org/10.36555/educare.v17i2.247>
- [4] M. & D. Selvi, "Aplikasi Media Pembelajaran Bahasa Inggris Menggunakan Unity 2D," *JUTINDA J. Inform. Unanda*, vol. 3, no. 2, hal. 40–45, 2024, [Daring]. Tersedia pada: <https://ojs.unanda.ac.id/index.php/jutinda>
- [5] F. F. Maulana dan N. Rochmawati, "Klasifikasi Citra Buah Menggunakan Convolutional Neural Network," *J. Informatics Comput. Sci.*, vol. 1, no. 02, hal. 104–108, 2019, doi: 10.26740/jinacs.v1n02.p104-108.
- [6] Ahmad Fariz Fuady, Dwiky Oldi Amsyah, Muhammad Farhan, Rusma Riansyah, dan M. Dayyan Dhiyaul Haq, "Implementasi Algoritma Convolutional Neural Network (CNN) untuk Pengenalan dan Klasifikasi Buah Berdasarkan Citra Digital," *J. Publ. Ilmu Komput. dan Multimed.*, vol. 4, no. 2, hal. 148–159, 2025, doi: 10.55606/jupikom.v4i2.4116.
- [7] R. A. Ines Heidiani Ikasari, Perani Rosyani, "Klasifikasi Jenis Buah Menggunakan Metode CNN," *J. Artif. Intell. Digit. Bus.*, vol. 4, no. 2, hal. 5451–5458, 2025.
- [8] E. Febrywinata, "Pengenalan Dan Klasifikasi Jenis Buah Menggunakan Metode CNN Secara Sederhana Dengan Menggunakan Google Colab," *Merkurius J. Ris. Sist. Inf. dan Tek. Inform.*, vol. 2, no. 4, hal. 185–193, 2024.
- [9] L. Huang, C. Yao, L. Zhang, S. Luo, F. Ying, dan W. Ying, "Enhancing computer image recognition with improved image algorithms," *Sci. Rep.*, vol. 14, no. 1, hal. 1–11, 2024, doi: 10.1038/s41598-024-64193-3.
- [10] T. Setyobudi, "APLIKASI PENGENALAN TOKOH WAYANG KULIT BERBASIS ANDROID DENGAN ALGORITMA CONVOLUTIONAL NEURAL NETWORK UNTUK ANAK SD," UNIVERSITAS PGRI SEMARANG, 2024.
- [11] M. S. Hawibowo, "Aplikasi Pengklasifikasi Kematangan Pepaya Menggunakan Metode Cnn Berbasis Android," 2024.
- [12] S. I. Komariah *et al.*, "Media Pembelajaran Pengenalan Buah (Fruits Zone) untuk Anak KB Menggunakan Deep Learning," *MIND (Multimedia Artif. Intell. Netw. Database) J.*, vol. 9, no. 1, hal. 13–24, 2024, [Daring]. Tersedia pada: <https://doi.org/10.26760/mindjournal.v9i1.13-24>
- [13] S. I. Komariah *et al.*, "Fruit Zone: Media Pembelajaran Interaktif Pengenalan Buah Anak Kelompok Belajar Menggunakan ResNet18," *Fakt. Exacta*, vol. 17, no. 1, hal. 20–29, 2024, doi: 10.30998/faktorexacta.v17i1.21101.
- [14] S. Salsabila dan Y. Gumala, "Kajian Literature : Media Pembelajaran Interaktif Di Sekolah Dasar," *Pediaqu J. Pendidik. Sos. dan Hum.*, vol. 4, no. 2, hal. 2706–2723, 2025, [Daring]. Tersedia pada: <https://publisherqu.com/index.php/pediaqu>
- [15] A. Z. K. F. & Q. M. ul H. Purwono, Alfian Ma'arif, Wahyu Rahmaniari, Haris Imam Karim Fathurrahman, "Understanding of Convolutional Neural Network (CNN): A Review," *Int. J. Robot. Control Syst.*, vol. 2, no. 4, hal. 739–748, 2022, doi: 10.31763/ijrcs.v2i4.888.
- [16] M. Sandler, A. Howard, M. Zhu, dan A. Zhmoginov, "MobileNetV2 : Inverted Residuals and Linear Bottlenecks," *Proc. IEEE Conf. Comput. Vis. pattern Recognit.*, hal. 4510–4520, 2018.
- [17] S. Bianco, R. Cadene, dan L. Celona, "Benchmark Analysis of Representative Deep Neural Network Architectures," *IEEE Access*, vol. 6, hal. 64270–64277, 2020, doi: 10.1109/ACCESS.2018.2877890.
- [18] B. Zoph, G. Brain, dan J. Shlens, "Learning Transferable Architectures for Scalable Image Recognition," *Proc. IEEE Conf. Comput. Vis. pattern Recognit.*, hal. 8697–8710, 2018.
- [19] A. C. Hartono dan A. R. Musliikh, "Penerapan Transfer Learning MobileNetV2 Pada Klasifikasi Citra Jenis Buah-Buahan Applying MobileNetV2 Transfer Learning for Image Classification of Fruit Types," *J. Inf. Syst. Appl. Dev.*, vol. 3, no. 2, hal. 103–111, 2025, doi: 10.26905/jisad.v3i2.16187.
- [20] A. Syakuroh, F. Monado, M. Ariani, dan E. Koriyanti, "ANALISIS AKURASI MODEL MOBILENETV2 DALAM KLASIFIKASI CITRA X-RAY UNTUK DETEKSI KONDISI PARU-PARU," *J. ONLINE Phys.*, vol. 10, no. 3, hal. 67–74, 2025.
- [21] S. S. Rambe, A. Asriyanik, dan P. Prajoko, "Penerapan Model Convolutional Neural Network ( Cnn ) Berbasis Mobilenetv2 Untuk," *JITET (Jurnal Inform. dan Tek. Elektro Ter.,* vol. 13, no. 3, 2025.