

## **PENERAPAN TEACHABLE MACHINE PADA KLASIFIKASI MACHINE LEARNING UNTUK IDENTIFIKASI TANAMAN HERBAL BERBASIS ANDROID DENGAN MENGGUNAKAN FLUTTER**

<sup>\*1</sup> *Deny Rochman Arifatno*, <sup>2</sup> *Mardana*, <sup>3</sup> *Rozi*

<sup>1,2</sup> *Fakultas Teknik Program Studi Teknik Informatika Universitas Cendekia Abditama*

<sup>3</sup> *Manajemen Informatika, STMIK Jakarta STI&K*

<sup>1,2</sup> *Jl. Islamic Raya, Klp. Dua, Kec. Klp. Dua, Kabupaten Tangerang, Banten 15811*

<sup>3</sup> *Jl. BRI Radio Dalam No.17, Jakarta Selatan*

<sup>\*1</sup> [deny@uca.ac.id](mailto:deny@uca.ac.id), <sup>2</sup> [danavgr55@gmail.com](mailto:danavgr55@gmail.com), <sup>3</sup> [roziborang72@gmail.com](mailto:roziborang72@gmail.com)

### **Abstrak**

*Pengenalan dan klasifikasi tanaman herbal merupakan aspek penting dalam botani dan kesehatan, tetapi sering memerlukan pengetahuan mendalam dan waktu. Penelitian ini bertujuan untuk menciptakan aplikasi Android yang menggunakan TensorFlow Lite dan Teachable Machine untuk mempermudah identifikasi dan klasifikasi tanaman herbal secara otomatis. Aplikasi ini dirancang untuk membantu pengguna dalam mengidentifikasi tanaman herbal hanya dengan menggunakan kamera ponsel pintar mereka. Metodologi penelitian dimulai dengan studi literatur terkait tanaman herbal dan teknik machine learning. Kemudian, data gambar tanaman herbal dikumpulkan dari berbagai sumber, termasuk koleksi digital dan gambar lapangan. Dataset ini kemudian digunakan untuk melatih model machine learning menggunakan TensorFlow Lite. Proses pelatihan model dilakukan dengan memanfaatkan platform Teachable Machine untuk mengoptimalkan akurasi dan efisiensi klasifikasi. Pengembangan aplikasi dilakukan dengan fokus pada antarmuka pengguna yang intuitif dan fungsionalitas yang memadai. Aplikasi ini diuji melalui uji coba BlackBox, di mana tidak ditemukan masalah signifikan. Evaluasi model menunjukkan performa tinggi dengan rata-rata akurasi gambar dari kamera sebesar 76% dan dari galeri sebesar 99%, menghasilkan total rata-rata keseluruhan akurasi sebesar 87,5%. Untuk memastikan keakuratan dan konsistensi hasil, setiap jenis tanaman, helai, atau dahan/ranting diuji dengan 20 kali pengambilan gambar. Pengujian berulang ini dirancang untuk mengatasi variabilitas dalam pencahayaan, sudut pengambilan gambar, dan kondisi lain yang dapat mempengaruhi hasil klasifikasi. Hasil ini menunjukkan bahwa aplikasi tidak hanya stabil tetapi juga efisien dalam memberikan akurasi tinggi dalam klasifikasi tanaman herbal. Keberhasilan penelitian ini menegaskan potensi aplikasi berbasis machine learning dalam mendukung identifikasi tanaman herbal, yang dapat memberikan manfaat besar bagi pengguna dalam bidang botani dan kesehatan herbal.*

**Kata Kunci:** *Aplikasi Android, Klasifikasi Tanaman, Machine Learning, Teachable Machine, TensorFlow Lite*

### **Abstract**

*The introduction and classification of herbal plants is an important aspect of botany and health, but it often requires in-depth knowledge and time. This research aims to create an Android application that uses TensorFlow Lite and Teachable Machine to make it easier to identify and classify herbal plants automatically. The app is designed to assist users in identifying herbal plants using only their smartphone camera. The research methodology began with the study of literature related to herbal plants and machine learning techniques. Then, the data of herbal plant images was collected from various sources, including digital collections and field images. This dataset is then used to train machine learning models using TensorFlow Lite. The model training process is carried out by utilizing the Teachable Machine platform to optimize the accuracy and efficiency of classification. The development of the app is done with a focus on an intuitive user interface and adequate functionality. The app was tested through a BlackBox trial, where no significant issues were found. The model evaluation showed high performance with an average image accuracy from the*

camera of 76% and from the gallery of 99%, resulting in an overall average of 87.5% accuracy. To ensure accuracy and consistency of results, each type of plant, strand, or branch/twig was tested with 20 shots. These iterative tests are designed to address variability in lighting, shooting angles, and other conditions that may affect classification results. These results show that the application is not only stable but also efficient in providing high accuracy in the classification of herbal plants. The success of this study confirms the potential of machine learning-based applications in supporting the identification of herbal plants, which can provide great benefits to users in the field of botany and herbal health.

**Key Words:** *Android Apps, Plant Classification, Machine Learning, Teachable Machine, TensorFlow Lite*

## PENDAHULUAN

Perkembangan teknologi *machine learning* telah membawa berbagai inovasi dalam pengembangan aplikasi *mobile*, terutama dalam hal analisis gambar. Salah satu teknologi yang mempermudah proses ini adalah *teachable Machine*, sebuah alat dari Google yang memungkinkan pengguna untuk melatih model *machine learning* tanpa memerlukan pengetahuan mendalam tentang pemrograman. Dengan menggunakan antarmuka yang mudah dipahami, pengguna dapat mengumpulkan data gambar, melatih model, dan mengeksportnya dalam format yang siap digunakan [1].

*TensorFlow Lite*, versi ringan dari *Tensor Flow*, dirancang khusus untuk menjalankan model *machine learning* pada perangkat *mobile* dengan efisien. Ini memungkinkan aplikasi untuk memproses data dan menghasilkan prediksi secara *real time* dengan penggunaan sumber daya yang minimal. *TensorFlow Lite* mengoptimalkan model sehingga dapat berjalan dengan baik pada perangkat dengan keterbatasan memori dan daya pemrosesan [2].

*Flutter*, *framework open-source* dari Google, memungkinkan pengembangan aplikasi *mobile* yang memiliki performa tinggi dan antarmuka pengguna yang menarik. Dengan *Flutter*, pengembang dapat menulis kode sekali dan menjalankannya di berbagai *platform*, termasuk Android dan iOS. *Flutter* mendukung integrasi berbagai teknologi,

termasuk *TensorFlow Lite*, sehingga memudahkan pembuatan aplikasi yang memanfaatkan model *machine learning* [3].

Dalam konteks penelitian ini, tujuan utama adalah mengembangkan aplikasi Android yang dapat mengidentifikasi tanaman herbal secara otomatis menggunakan teknologi *machine learning*. Aplikasi ini dibangun dengan *Flutter* dan memanfaatkan model *machine learning* yang dilatih menggunakan *Teachable Machine*. Model ini kemudian dikonversi ke format *TensorFlow Lite* dan diintegrasikan ke dalam aplikasi. Dengan pendekatan ini, diharapkan aplikasi dapat memberikan solusi praktis dan efisien untuk klasifikasi gambar tanaman herbal di perangkat *mobile*, membantu pengguna dalam identifikasi tanaman herbal secara cepat dan akurat [5].

Penerapan teknologi ini tidak hanya meningkatkan kemampuan aplikasi dalam mengklasifikasikan gambar, tetapi juga membuka peluang untuk penggunaan *machine learning* dalam aplikasi *mobile* lainnya yang memerlukan analisis gambar secara *real-time*.

*Teachable Machine* merupakan sebuah alat *online* dari Google yang memungkinkan pengguna untuk melatih model *machine learning* dengan mudah tanpa harus menulis kode. Dengan antarmuka yang sederhana dan *user-friendly*, *teachable machine* memungkinkan pengguna untuk mengunggah gambar, mengelompokkan gambar ke dalam kategori yang berbeda,

dan melatih model untuk mengenali pola dalam gambar tersebut. Model yang di hasilkan dapat digunakan untuk berbagai aplikasi, seperti klasifikasi gambar dan deteksi objek. Setelah pelatihan selesai, model dapat diekspor dalam format *tensorflow lite*, yang memungkinkan integrasi dengan aplikasi mobile [4].

*TensorFlow Lite* merupakan versi ringan dari *Tensor Flow*, sebuah *framework machine learning* yang dikembangkan oleh Google. *TensorFlow Lite* di rancang khusus untuk menjalankan model *machine learning* pada perangkat *mobile* dan *embedded* dengan efisiensi tinggi. Ini mengoptimalkan model agar berjalan cepat dan hemat sumber daya di perangkat dengan kemampuan pemrosesan terbatas, seperti *smartphone* dan tablet. *TensorFlow Lite* mendukung berbagai jenis operasi *machine learning* dan menyediakan *API* untuk integrasi dengan aplikasi *mobile*, memungkinkan aplikasi untuk melakukan inferensi model dengan responsivitas yang baik [2].

*Flutter* adalah *framework open-source* yang dikembangkan oleh Google untuk membangun aplikasi *mobile* dan web dari satu basis kode yang sama. Dengan menggunakan *Dart* sebagai bahasa pemrograman, *Flutter* memungkinkan pengembang untuk membuat aplikasi yang memiliki performa tinggi dan antarmuka pengguna yang menarik. Salah satu keunggulan *Flutter* adalah kemampuannya untuk membangun aplikasi yang dapat berjalan di berbagai *platform* dengan kode yang sama, termasuk *Android* dan *iOS*. *Flutter* juga mendukung integrasi dengan berbagai teknologi, termasuk *TensorFlow Lite*, sehingga memudahkan pengembangan aplikasi yang memanfaatkan model *machine learning* [3].

## METODE PENELITIAN

Metode penelitian ini terdiri dari lima langkah utama: studi literatur, pengumpulan data, pelatihan model, pengembangan aplikasi, dan evaluasi.

1. Studi Literatur adalah langkah awal yang dilakukan untuk memahami dasar-dasar teknologi yang akan digunakan dalam penelitian. Pada tahap ini, dilakukan tinjauan mendalam terhadap alat-alat seperti *teachable machine*, *tensorflow lite*, dan *flutter*. Fokus utamanya adalah memahami cara kerja masing-masing teknologi serta potensi integrasinya dalam pengembangan aplikasi *mobile*. Selain itu, analisis penelitian terdahulu mengenai klasifikasi gambar menggunakan *machine learning* dan penerapan teknologi serupa juga dilakukan untuk mendapatkan wawasan yang lebih luas.
2. Pengumpulan Data merupakan langkah penting yang dilakukan untuk melatih model *machine learning*. Data yang diperlukan berupa gambar-gambar tanaman herbal, yang dikumpulkan dari berbagai sumber untuk memastikan keberagaman. Dalam penelitian ini, data gambar utama diperoleh melalui dua metode. Pertama, data gambar sekunder diambil dari dataset yang tersedia di *website* Kaggle, khususnya dari [Deteksi Tanaman Herbal Model H5 & TFLite \(kaggle.com\)](#), yang menyediakan berbagai gambar tanaman herbal yang telah dilabeli. Kedua, data gambar primer dikumpulkan melalui observasi langsung, dengan mengambil gambar tanaman herbal menggunakan ponsel Xiaomi dengan kamera 64 MP di lingkungan sekitar. Proses pengambilan gambar secara langsung ini bertujuan untuk mendapatkan variasi gambar yang lebih sesuai dengan kondisi nyata di lapangan. Semua gambar yang dikumpulkan, baik dari sumber sekunder maupun primer, kemudian diberi label sesuai dengan jenis tanaman herbal yang

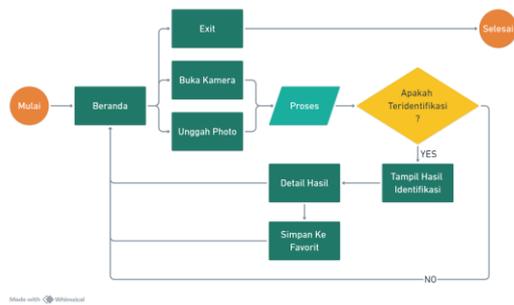
- ada. Proses labeling ini penting untuk memastikan model dapat belajar membedakan berbagai kategori tanaman secara akurat.
3. Pelatihan Model dengan *teachable machine* dilakukan setelah data terkumpul. Proses ini dimulai dengan mengunggah gambar-gambar yang telah dilabeli ke *platform teachable machine*. Melalui *platform* ini, model dilatih menggunakan data yang telah diunggah untuk mengenali pola dan fitur spesifik dari setiap kategori tanaman. Setelah proses pelatihan selesai, model diekspor dalam format *TensorFlow Lite (.tflite)*, yang dirancang untuk berjalan secara efisien pada perangkat *mobile*.
  4. Pengembangan Aplikasi melibatkan pembuatan aplikasi Android menggunakan *Flutter* dan integrasi model *tensorflow lite* ke dalam aplikasi tersebut. Desain aplikasi dilakukan dengan menggunakan *Flutter* untuk menciptakan antarmuka pengguna yang mudah dipahami dan menarik. Selanjutnya, *tensorflow lite* diintegrasikan ke dalam aplikasi dengan menambahkan dependensi yang diperlukan dan memuat model *.tflite*. Implementasi logika inferensi dilakukan dengan mengembangkan kode untuk menangkap gambar dari kamera atau galeri, memproses gambar tersebut dengan model *tensorflow lite*, dan menampilkan hasil klasifikasi kepada pengguna.
  5. Evaluasi dilakukan untuk menilai kinerja aplikasi dan akurasi model. Pengujian fungsional dilakukan pada berbagai perangkat Android untuk memastikan bahwa aplikasi berjalan dengan baik dan semua fitur berfungsi sesuai yang diharapkan. Evaluasi akurasi model dilakukan dengan menguji model menggunakan dataset

gambar baru dan membandingkan hasil klasifikasi dengan label yang benar untuk menentukan tingkat akurasi. Selain itu, uji responsivitas dilakukan untuk memastikan aplikasi memberikan hasil klasifikasi dalam waktu yang dapat diterima, yaitu kurang dari satu detik per gambar.

### Analisa Perancangan Sistem *Flowchart Diagram*

Sebelum melangkah ke tahap implementasi sistem, ada satu langkah penting yang harus dilakukan terlebih dahulu, yaitu membuat *flowchart* sistem. *Flowchart* ini bukan sekadar diagram, *flowchart* merupakan representasi visual dari prosedur yang menggambarkan alur kerja atau tahapan-tahapan yang harus dilalui oleh suatu program. Pembuatan *flowchart* sangatlah krusial karena berfungsi sebagai panduan bagi analis dalam mengidentifikasi dan menyelesaikan masalah yang mungkin muncul selama pengembangan. *Flowchart* adalah representasi prosedural yang menunjukkan langkah-langkah atau urutan kerja suatu program. *Flowchart* ini berperan penting dalam membantu analis menyelesaikan masalah. Dengan adanya *flowchart*, penyelesaian masalah yang berkaitan dengan debugging pada aplikasi atau situs web dapat dilakukan dengan lebih efisien [6].

Di bawah ini terdapat gambar *flowchart* yang menggambarkan alur aplikasi dari awal tampilan hingga proses akhir. *Flowchart* ini memberikan visualisasi yang rinci mengenai setiap langkah yang dilalui oleh pengguna dan bagaimana aplikasi merespons pada setiap tahapannya. Mulai dari membuka aplikasi hingga tindakan akhir, *flowchart* ini berfungsi sebagai panduan yang memudahkan pemahaman mengenai seluruh proses kerja aplikasi, baik bagi pengembang maupun pengguna.

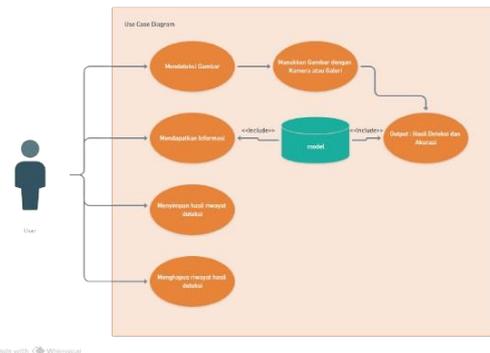


Gambar 1. Flowchart Diagram

### Use Case Diagram

*Use Case Diagram* merupakan alat penting dalam pengembangan sistem yang digunakan untuk mengidentifikasi dan menggambarkan interaksi antara pengguna (aktor) dan sistem dalam berbagai skenario. Diagram ini memberikan visualisasi yang jelas tentang bagaimana pengguna akan berinteraksi dengan sistem, termasuk berbagai fungsi atau layanan yang dapat diakses oleh pengguna [7].

Gambar di bawah ini adalah *use case diagram* yang menggambarkan aktivitas-aktivitas yang terjadi dalam penggunaan aplikasi ini. Diagram tersebut menjelaskan interaksi antara pengguna dengan berbagai fungsi yang tersedia di dalam aplikasi yaitu: Pengguna dapat mengunggah gambar tanaman melalui kamera atau galeri. Gambar tersebut akan diproses oleh model deteksi aplikasi yang menggunakan algoritma pembelajaran mesin untuk mengidentifikasi dan mengklasifikasikan tanaman, serta menampilkan hasil deteksi beserta tingkat akurasi. Pengguna juga dapat melihat informasi detail tentang jenis tanaman yang terdeteksi. Selain itu, aplikasi memiliki fitur untuk menyimpan riwayat deteksi, memungkinkan pengguna untuk mengakses kembali hasil deteksi sebelumnya. Pengguna juga dapat menghapus riwayat deteksi yang tidak diperlukan, sehingga data tetap terorganisir dengan baik.

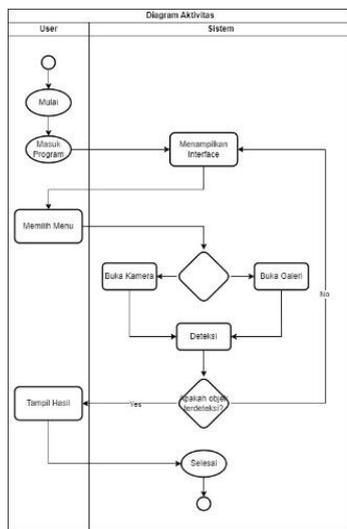


Gambar 2. Use Case Diagram

### Activity Diagram

*Activity Diagram* adalah salah satu jenis diagram dalam *Unified Modeling Language (UML)* yang berfungsi untuk menggambarkan aliran proses bisnis atau aktivitas dalam sebuah sistem. Diagram ini memudahkan pengembang perangkat lunak untuk memodelkan proses bisnis atau alur kerja sistem dengan cara yang lebih terstruktur dan sistematis [7].

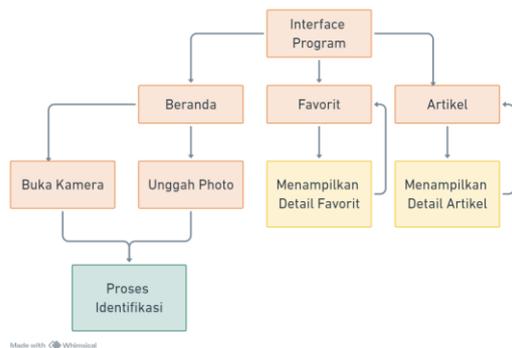
Diagram aktivitas ini menggambarkan alur proses interaksi antara pengguna dan sistem dalam sebuah aplikasi. Proses dimulai ketika pengguna membuka aplikasi dan masuk ke dalam program, yang kemudian diikuti oleh sistem menampilkan antarmuka (*interface*) aplikasi. Di sini, pengguna dapat memilih menu yang tersedia, yang mengarahkan mereka ke opsi untuk membuka kamera guna mengambil foto atau membuka galeri untuk memilih foto yang sudah ada. Setelah pengguna memilih salah satu dari opsi tersebut, sistem melanjutkan ke tahap deteksi objek dalam gambar yang diambil atau diunggah. Jika sistem berhasil mendeteksi objek dalam gambar, hasil deteksi akan ditampilkan kepada pengguna. Namun, jika objek tidak terdeteksi, proses berakhir tanpa menampilkan hasil apapun. Diagram ini menggambarkan secara jelas bagaimana alur interaksi antara pengguna dan sistem berjalan, mulai dari inisiasi proses hingga penyelesaian, baik dalam skenario deteksi berhasil maupun tidak.



Gambar 3. Activity Diagram

### Struktur Navigasi

Struktur Navigasi adalah tahapan dalam sebuah program yang menggambarkan bagaimana pengguna berpindah antar bagian [8]. Struktur navigasi yang diterapkan dalam penulisan ini adalah model campuran yang terlihat pada Gambar 4.



Gambar 4. Struktur Navigasi

Struktur Navigasi aplikasi ini dirancang untuk memberikan kemudahan akses bagi pengguna melalui antarmuka utama yang menghubungkan ke tiga fitur utama: Beranda, Favorit, dan Artikel. Dari Beranda, pengguna memiliki opsi untuk membuka kamera guna mengambil foto langsung atau mengunggah foto yang sudah ada dari galeri mereka. Setelah itu, foto akan diproses untuk identifikasi lebih lanjut. Selain itu, pengguna dapat mengakses halaman Favorit, di mana mereka dapat melihat dan menampilkan detail dari item yang telah mereka tandai sebagai favorit. Fitur Artikel memungkinkan pengguna

untuk menjelajahi berbagai artikel yang tersedia, dengan opsi untuk melihat lebih dalam rincian dari artikel tertentu. Dengan struktur ini, pengguna dapat dengan mudah berpindah dari satu fitur ke fitur lainnya, memastikan pengalaman yang lancar dan intuitif dalam menggunakan aplikasi.

## HASIL DAN PEMBAHASAN

### Implementasi

Implementasi sistem merupakan implementasi hasil rancangan antarmuka ke dalam sistem yang dibangun menggunakan framework flutter dengan bahasa pemrograman Dart. Berikut implementasi dari setiap fungsional pada aplikasi identifikasi tanaman herbal.

#### 1. Antarmuka *Flashscreen*



Gambar 5. Antarmuka *Flashscreen*

Gambar 5 menampilkan *flashscreen* saat pengguna membuka aplikasi. Tujuan dari halaman ini adalah untuk memberikan pengalaman awal yang menyenangkan dan memberikan identitas visual kepada pengguna bahwa aplikasi sedang dimuat. Pada halaman ini, menampilkan logo dan nama aplikasi.

#### 2. Antarmuka *Homepage*



Gambar 6. Antarmuka *Homepage*

Gambar 6 menampilkan *homepage* atau menu utama yang terdiri dari buka kamera, unggah gambar, button exit untuk keluar dari aplikasi dan navigasi untuk perpindahan ke halaman Beranda, Favorit dan Artikel.

### 3. Antarmuka *ShowlogDialog*



Gambar 7. Antarmuka *ShowlogDialog*

Gambar 7 menampilkan *ShowlogDialog* atau pemberitahuan bahwa objek terdeteksi atau tidak, jika objek terdeteksi akan menampilkan informasi nama tanaman dan hasil akurasi tanaman tersebut.

### 4. Antarmuka *Detail*



Gambar 8. Antarmuka *Detail*

Gambar 8 menampilkan informasi deskripsi mengenai tanaman herbal setelah proses identifikasi selesai.

### 5. Antarmuka *Daftar Favorit*



Gambar 9. Antarmuka *Daftar Favorit*

Gambar 9 menampilkan daftar tanaman herbal yang telah ditandai sebagai favorit oleh pengguna. Pengguna dapat dengan mudah mengakses dan mengelola daftar tanaman yang diminatinya melalui halaman ini. Tampilan halaman ini berisi daftar gambar atau nama tanaman dengan opsi untuk menghapus atau memperbarui status favorit.

### 6. Antarmuka *Detail Favorit*



Gambar 10. Antarmuka *Detail Favorit*

Gambar 10 menampilkan informasi deskripsi mengenai tanaman herbal yang telah ditandai sebagai favorit oleh pengguna. mirip dengan antarmuka detail.

### 7. Antarmuka *Daftar Artikel*



Gambar 11. Antarmuka Daftar Artikel

Gambar 11 menampilkan daftar artikel atau terkait dengan tanaman herbal. Pengguna dapat menjelajahi berbagai artikel yang tersedia untuk mendapatkan informasi lebih lanjut tentang tanaman, penggunaannya, atau berita terkini seputar herbalisme.

8. Antarmuka Detail Artikel



Gambar 12. Antarmuka Detail Artikel

Gambar 12 menampilkan informasi lengkap dari sebuah artikel, seperti gambar tanaman, jenis tanaman, nama ilmiah dari tanaman, tipe tanaman, deskripsi tanaman, kandungan dan manfaat dari tanaman herbal.

**Pengujian BlackBox**

Pengujian *BlackBox* yang dilakukan pada aplikasi ini dapat dilihat pada di bawah ini:

Tabel 1. Pengujian Aplikasi Menggunakan *BlackBox*

No	Fungsi	Hasil Yang Diharapkan
1	Menampilkan Halaman <i>Flash screen</i>	Berhasil Menampilkan
2	Menampilkan Halaman	Berhasil Menampilkan

	<i>Homepage</i>	
3	Menampilkan Kamera	Berhasil Menampilkan
4	Menampilkan Galeri	Berhasil Menampilkan
5	Menampilkan <i>Showdialog</i>	Berhasil Menampilkan
6	Menampilkan Halaman Daftar Favorit	Berhasil Menampilkan
7	Menampilkan Halaman Detail Favorit	Berhasil Menampilkan
8	Menampilkan Halaman Daftar Artikel	Berhasil Menampilkan
9	Menampilkan Halaman Detail Artikel	Berhasil Menampilkan

Pengujian *BlackBox* dilakukan untuk menilai bagaimana aplikasi Deteksi Tanaman Herbal berfungsi secara keseluruhan. Berdasarkan hasil pengujian, aplikasi ini berhasil melewati semua tes yang mencakup fitur-fitur utama. Pada pengujian ini, semua tur yang diuji berhasil beroperasi dengan baik. Misalnya, aplikasi dapat menampilkan halaman *flashscreen*, halaman beranda, mengaktifkan fungsi kamera dan galeri, menampilkan *showlogdialog*, halaman daftar Favorit, detail Favorit, halaman daftar artikel dan detail artikel. Semua tombol navigasi dan interaksi pengguna berjalan lancar tanpa ada masalah signifikan.

**Evaluasi Model**

*Accuracy Per Class* adalah metrik evaluasi yang menghitung proporsi instance yang di klasifikasikan dengan benar untuk setiap kelas tertentu. Ini membantu dalam memahami seberapa baik model bekerja untuk masing-masing kelas secara individual.

CLASS	ACCURACY	# SAMPLES
Daun Belimbing Wul...	1.00	77
Daun Bidara	1.00	45
Daun Binahong	1.00	35
Daun Ceremai	1.00	51
Daun Dadap	1.00	47
Daun Jambu Biji	1.00	51
Daun Kelor	1.00	45
Daun kunyit	1.00	36
Daun Mengkudu	1.00	39
Daun Pandan	1.00	31
Daun Pepaya	1.00	31
Daun Seledri	1.00	73
Daun Sirih	1.00	28
Daun Sirsak	1.00	48
Lidah Buaya	1.00	21

Gambar 13. Accuracy Per Class

Tabel yang ditampilkan mencerminkan hasil dari proses *training* model menggunakan *teachable machine* dengan parameter *training* sebagai berikut: *epochs* sebanyak 50, *batch size* sebesar 128, dan *learning rate* 0.001.

### Pengujian Akurasi Model

Pengujian Akurasi Model yang dilakukan pada aplikasi di handphone xiaomi redmi note 10s ini dapat dilihat pada tabel di bawah ini:

No	Jenis Tanaman	Metode	Akurasi (%)														
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	Daun Bidara	hasil galeri	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
		hasil kamera	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
		Rata-rata	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100

Gambar 14. Kumpulan Data Hasil Dari Pengujian Akurasi Model

Hasil dari gambar tabel di atas menunjukkan kumpulan data yang diperoleh dari pengujian akurasi model dalam mendeteksi tanaman herbal. Pengujian dilakukan sebanyak 20 kali pada setiap jenis tanaman terutama pada

helai dan tangkai atau sebagian dari tanaman tersebut, untuk memastikan bahwa data yang diperoleh adalah *valid* dan representatif. Teknik pengambilan gambar meliputi dua metode utama. Pertama, foto diambil secara langsung dari objek tanaman herbal, yang meliputi pengambilan gambar dari 1 helai daun dan 1 tangkai tanaman. Pengambilan gambar ini dilakukan menggunakan kamera untuk menilai kemampuan model dalam mendeteksi tanaman dalam kondisi nyata dan variatif.

No	hasil galeri		No	hasil kamera	
	h	t		h	t
1	99,2	99,2	1	85,25	73,1
2	99,2	99,2	2	85,25	77,45
3	99,2	99,2	3	59,2	73,05
4	99,25	99,2	4	87,4	92,75
5	99,25	99,2	5	56,45	89,4
6	97,2	99,2	6	94,55	94,1
7	99,25	99,2	7	42,7	58,75
8	99,2	99,2	8	49,5	92,5
9	98,65	99,2	9	91,1	70,3
10	99,25	99,2	10	75,65	79,8
11	99,2	99,2	11	70,6	66,95
12	99,2	99,2	12	59,2	92,5
13	99,2	99,2	13	88,9	88,85
14	99,15	99,2	14	48,85	74,65
15	99,2	99,2	15	73,7	89,45
Rata-rata	99,04	99,2	Rata-rata	71,22	80,9067

Gambar 15. Rata-rata Deteksi Hasil Pengujian Model Dari Kamera dan Galeri

Hasil dari kedua tabel di atas menunjukkan pengelompokan data hasil rata rata dari pengujian akurasi model dalam mendeteksi 15 jenis data tanaman herbal, yang dilakukan sebanyak 20 kali percobaan untuk setiap jenis data. Data ini terbagi menjadi dua kategori utama: gambar dari galeri dan gambar yang diambil menggunakan kamera. Dari hasil pengujian pada gambar yang diambil dari galeri, diperoleh nilai rata-rata akurasi sebagai berikut: untuk gambar 1 helai daun, model mencapai rata-rata akurasi sebesar 99,04, sementara untuk gambar tangkai tanaman, rata-rata akurasinya adalah 99,2. Nilai-nilai ini menunjukkan bahwa model memiliki performa yang sangat baik dalam mengenali tanaman herbal dari dataset galeri yang telah ada.

Sebaliknya, untuk gambar yang diambil langsung menggunakan kamera, hasilnya menunjukkan rata-rata akurasi yang lebih rendah. Untuk gambar 1 helai daun, model memperoleh rata-rata akurasi sebesar 71,22, dan untuk gambar tangkai

tanaman, rata-rata akurasi adalah 80,90666667. Penurunan akurasi ini mungkin disebabkan oleh variasi dalam kondisi pencahayaan, sudut pengambilan gambar, atau faktor lain yang mempengaruhi kualitas gambar yang diambil secara langsung.

Untuk menghitung rata-rata hasil pengujian akurasi dari kamera dan galeri:

$$\text{Rata-rata} = \frac{a}{b}$$

Keterangan

a = Jumlah Total Nilai

b = Jumlah Data

1. Gambar dari kamera:
  - a) Rata-rata nilai akurasi dengan teknik 1 helai daun: 71,22
  - b) Rata-rata nilai akurasi dengan teknik 1 tangkai atau sebagian dari tanaman: 80,90666667

Rata-rata akurasi gambar dari hasil kamera:

$$= \frac{71,22 + 80,90666667}{2} = \frac{152,12666667}{2} = 76,06333333$$

Jadi, rata-rata keseluruhan hasil deteksi dari kamera jika dibulatkan ke angka bulat terdekat adalah 76%.

2. Gambar dari Galeri:
  - a) Rata-rata nilai akurasi dengan teknik 1 helai daun: 99,04%
  - b) Rata-rata nilai akurasi dengan teknik 1 tangkai atau sebagian dari tanaman: 99,2%

Rata-rata akurasi gambar dari hasil galeri:

$$= \frac{99,04\% + 99,2\%}{2} = \frac{198,24\%}{2} = 99,12\%$$

Jadi, rata-rata keseluruhan hasil deteksi dari kamera jika dibulatkan ke angka bulat terdekat adalah 99%.

3. Total rata-rata keseluruhan akurasi
  - a) Rata-rata akurasi kamera: 76%
  - b) Rata-rata akurasi galeri: 99%

Total rata-rata keseluruhan:

$$= \frac{76\% + 99\%}{2} = \frac{175\%}{2} = 87,5\%$$

Nilai-nilai ini menunjukkan bahwa model memiliki performa yang sangat baik dalam mengenali dan mengklasifikasikan tanaman herbal, baik

dari gambar yang diambil langsung menggunakan kamera maupun dari gambar yang diambil dari galeri. Total rata-rata keseluruhan akurasi yang diperoleh adalah 87,5%, yang mencerminkan kehandalan dan ketepatan model dalam klasifikasi tanaman herbal.

## KESIMPULAN DAN SARAN

Aplikasi Deteksi Tanaman Herbal yang telah dikembangkan berhasil mengintegrasikan teknologi *Machine Learning* dari *TensorFlow Lite* dan *Teachable Machine* ke dalam platform Android. Aplikasi ini dirancang untuk memudahkan pengguna dalam mengenali dan mengelompokkan tanaman herbal melalui antarmuka yang mudah digunakan dan responsif, serta memiliki integrasi yang baik dengan kamera untuk pengambilan foto langsung atau dari galeri. Model *machine learning* yang diterapkan mampu melakukan prediksi dengan cepat dan akurat.

Berdasarkan hasil uji coba BlackBox menunjukkan bahwa aplikasi berfungsi dengan baik tanpa masalah signifikan. Semua fitur utama, seperti layar utama, navigasi, kamera, galeri, dan manajemen daftar favorit, beroperasi secara lancar, dan interaksi pengguna dengan tombol serta fitur lainnya berjalan dengan baik. Evaluasi model juga menunjukkan hasil yang sangat positif, dengan nilai keseluruhan mencapai 1.00 tanpa adanya indikasi *overfitting*. Pengujian akurasi model mengungkapkan performa yang sangat baik, dengan rata-rata akurasi gambar dari kamera mencapai 76%, rata-rata akurasi gambar dari galeri mencapai 99%, dan total rata-rata akurasi keseluruhan mencapai 87,5%. Dengan demikian, aplikasi ini tidak hanya berfungsi dengan optimal tetapi juga memberikan hasil akurasi yang tinggi dalam pengolahan gambar dari berbagai sumber, menandakan keberhasilan dalam pengembangan dan implementasinya.

Saran untuk pengembangan kedepannya untuk penelitian.

1. Peningkatan Performa Model: Secara rutin mengevaluasi dataset pelatihan untuk meningkatkan keakuratan model dalam mengidentifikasi tanaman yang lebih kompleks atau dalam berbagai kondisi lingkungan.
  2. Perluasan Fitur Aplikasi: Menambahkan fitur baru seperti memberikan rekomendasi tanaman berdasarkan lokasi atau musim, dan menghubungkan dengan platform pembelajaran untuk membantu pengguna mempelajari lebih banyak tentang tanaman herbal.
  3. Optimalisasi Antarmuka Pengguna: Meningkatkan tampilan aplikasi dengan perbaikan desain yang lebih mudah dipahami dan menambahkan pemberitahuan yang lebih bermanfaat.
  4. Peningkatan Dokumentasi dan Bantuan: Berikan informasi yang lengkap dan jelas serta petunjuk yang mudah dipahami agar pengguna bisa menggunakan semua fitur aplikasi dengan baik.
- web, and desktop from a single codebase”, 2023. [Online]. Available: <https://flutter.dev> [Accessed Sep. 16, 2024].
- [4] Google AI, “Teachable Machine: Train a model to recognize your own images, sounds, and poses, [Online]. Available: <https://teachablemachine.withgoogle.com> [Accessed Sep. 16, 2024].
- [5] Simbolon, S. P., & Maulany, R., “Perancangan Aplikasi Pendeteksi Jenis-jenis Sampah Berbasis Android: Development of an Android-Based Waste Type Detection”, MALCOM: Indonesian Journal of Machine Learning and Computer Science, 2024, 4(3), 926-935. [Accessed Sep. 16, 2024].
- [6] Ronaldo, M. D., & Maziidi, I., “SISTEM PAKAR DIAGNOSIS PENYAKIT TANAMAN CABAI MENGGUNAKAN METODE CERTAINTY FACTOR BERBASIS ANDROID”, Jurnal Kecerdasan Buatan dan Teknologi Informasi, 2024, 3(3), 98-106. [Accessed Sep. 16, 2024].

#### DAFTAR PUSTAKA

- [1] Chazar, C & Rafsanjani, M. H, Penerapan Teachable Machine Pada Klasifikasi Machine Learning Untuk Identifikasi Bibit Tanaman, Seminar Nasional : Inovasi dan Adopsi Teknologi Metaverse Is The Future Of Work, 21 Mei 2002. [Accessed Sep. 16, 2024].
- [2] TensorFlow Team, “TensorFlow Lite Model Optimization Toolkit”, 2022. [Online]. Available: <https://www.tensorflow.org/lite/guide/optimization> [Accessed Sep. 16, 2024].
- [3] Google, “Flutter: Build natively compiled applications for mobile, web, and desktop from a single codebase”, 2023. [Online]. Available: <https://flutter.dev> [Accessed Sep. 16, 2024].
- [4] Google AI, “Teachable Machine: Train a model to recognize your own images, sounds, and poses, [Online]. Available: <https://teachablemachine.withgoogle.com> [Accessed Sep. 16, 2024].
- [5] Simbolon, S. P., & Maulany, R., “Perancangan Aplikasi Pendeteksi Jenis-jenis Sampah Berbasis Android: Development of an Android-Based Waste Type Detection”, MALCOM: Indonesian Journal of Machine Learning and Computer Science, 2024, 4(3), 926-935. [Accessed Sep. 16, 2024].
- [6] Ronaldo, M. D., & Maziidi, I., “SISTEM PAKAR DIAGNOSIS PENYAKIT TANAMAN CABAI MENGGUNAKAN METODE CERTAINTY FACTOR BERBASIS ANDROID”, Jurnal Kecerdasan Buatan dan Teknologi Informasi, 2024, 3(3), 98-106. [Accessed Sep. 16, 2024].
- [7] Pranoto S, Sutiono S, Nasution D., “Penerapan UML Dalam Perancangan Sistem Informasi Pelaporan Dan Evaluasi Pembangunan Pada Bagian Administrasi Pembangunan Sekretariat Daerah Kota Tebing Tinggi”, Surplus: Jurnal Ekonomi dan Bisnis, 2024 Jun 24;2(2):384-401. [Accessed Sep. 16, 2024].
- [8] Rahayu Noveandini, Maria Sri Wulandari, and Sri Handayani., “Pengembangan Media Belajar Siswa Untuk Pengenalan Negara Negara Asean Berbasis Android”, Jurnal SIKOMTEK, vol. 14, no. 2, July 2024, pp. 158-63. [Accessed Sep. 16, 2024].