

IMPLEMENTASI METODE CONVOLUTIONAL NEURAL NETWORK UNTUK KLASIFIKASI RAS ANJING DENGAN ARSITEKTUR INCEPTIONv3

Gracesella Claren¹, Genrawan Hoendarto², Susana³

¹²³Program Studi Informatika, Fakultas Teknologi Informasi Universitas Widya Dharma Pontianak
e-mail: ¹gracela.claren1@gmail.com, ²genrawan@widyadharm.ac.id, ³susana@widyadharm.ac.id

Abstract

This research aims to develop an application capable of identifying dog breeds through graphic image detection using the Convolutional Neural Network (CNN) method. The implementation of artificial intelligence (AI) technology is increasingly widespread, including for pet breed recognition applications. The main challenge in conventional dog breed identification lies in the complexity of their physical variations, fur, and colors, requiring specific knowledge in the field of domestic animals for accurate results. By utilizing the CNN-based deep learning algorithm in the development of this AI application, accurate recognition of unique patterns and characteristics that distinguish each dog breed can be achieved through proper training with a large and comprehensive dataset. In this research, the model was trained on 2,758 images with an 80:20 dataset ratio, divided into 15 classes using the Inceptionv3 model architecture. During the training process, the model successfully achieved an F-1 Score above 85% for 13 out of 15 classes, with an overall final accuracy of 94.71%. The conclusion obtained is that the desktop-based dog breed recognition application works optimally in identifying dog breeds.

Keywords—Artificial Intelligence, Convolutional Neural Network, Dog Breed

Penelitian ini bertujuan untuk mengembangkan aplikasi yang mampu mengidentifikasi ras anjing melalui deteksi gambar grafis menggunakan metode Convolutional Neural Network (CNN). Implementasi teknologi kecerdasan buatan (AI) semakin meluas, termasuk untuk aplikasi pengenalan ras hewan peliharaan. Masalah utama dalam identifikasi ras anjing konvensional terletak pada kompleksitas variasi fisik, bulu, dan warna anjing, yang membutuhkan pengetahuan khusus di bidang hewan peliharaan untuk mendapatkan hasil yang akurat. Dengan memanfaatkan algoritma deep learning berbasis CNN dalam pengembangan aplikasi AI ini, pengenalan pola dan karakteristik unik yang membedakan setiap ras anjing dapat dicapai secara akurat melalui pelatihan yang tepat dengan dataset yang besar dan lengkap. Pada penelitian ini, model dilatih pada 2.758 citra dengan perbandingan dataset 80:20 yang terbagi ke dalam 15 kelas menggunakan arsitektur model Inceptionv3. Dalam proses pelatihan, model berhasil mencapai F-1 Score diatas 85% untuk 13 kelas dari 15 kelas, serta tingkat akurasi akhir keseluruhan 94,71%. Kesimpulan yang diperoleh adalah aplikasi pengenalan ras anjing berbasis desktop bekerja secara optimal dalam mengidentifikasi ras anjing.

Keywords—Kecerdasan Buatan, Jaringan Saraf Konvolusi, Ras Anjing

1. PENDAHULUAN

Saat ini, dunia tengah mengalami transformasi besar dalam bidang teknologi informasi yang dikenal sebagai revolusi industri 4.0 menuju *society* 5.0. Revolusi ini ditandai dengan terjadinya komputerisasi dan perubahan fundamental dalam berbagai aspek pekerjaan manusia dari manual menjadi digital. Revolusi industri 4.0 telah mengubah gaya hidup dan cara bekerja manusia secara signifikan dengan adanya integrasi sistem komputer secara digital pada mesin (*embedded system*). Pemanfaatan teknologi seperti *Artificial Intelligence* (AI) tidak hanya membawa kemajuan di bidang industri dan manufaktur, tetapi juga hampir semua bidang yang mencakup ekonomi, politik, sosial, budaya, dan lainnya.

Pengembangan AI menjadi salah satu landasan utama implementasi sistem digital di masyarakat luas. Salah satu metode logika AI yang populer digunakan dalam penerapan *society* 5.0 adalah *Convolutional Neural Network* (CNN). Saat ini, metode *deep learning* dengan algoritma CNN telah mendominasi dunia pemrosesan citra di ranah kecerdasan buatan. Keunggulan CNN terletak pada kemampuannya dalam memahami pola kompleks dalam data visual, terutama gambar. Hal ini memungkinkan program AI berbasis CNN untuk mengenali objek dalam gambar dengan akurasi yang tinggi.

Pengenalan ras anjing (*Canis domesticus*) merupakan salah satu contoh penerapan nyata dari teknologi AI yang dapat membantu manusia memahami dunia hewan domestik, khususnya anjing. Penggunaan aplikasi pengenalan ras anjing berbasis pengenalan citra tidak hanya memberikan hiburan dan pengetahuan umum, tetapi juga membuka peluang untuk edukasi yang lebih mendalam tentang berbagai ras anjing domestik. Secara umum, kendala utama dalam pengenalan ras anjing adalah kompleksitas variasi fisik, bulu, dan warna yang dimiliki berbagai ras anjing. Dengan memanfaatkan algoritma *deep learning* berbasis CNN dalam pembuatan aplikasi AI untuk mengenali karakteristik dan ciri kompleks pada setiap ras anjing, permasalahan ini dapat diatasi. Penelitian ini bertujuan untuk menghasilkan aplikasi yang mampu mengidentifikasi ras anjing melalui deteksi citra grafis dengan menggunakan metode CNN.

2. METODE PENELITIAN

2.1 Metodologi Penelitian

2.1.1 Studi Literatur

Peneliti mempelajari literatur-literatur yang meliputi laporan penelitian, jurnal ilmiah, buku, ensiklopedia, e-book yang diunduh dari internet and jenis literatur lainnya yang berkaitan dengan topik yang dibahas, seperti *AI, CNN, Machine Learning, Image Processing*, ras anjing dan lain-lain.

2.1.2 Pengumpulan Data

Peneliti mengumpulkan data yang akan digunakan untuk 3 tahap perancangan model, yakni pelatihan, validasi, dan pengujian. Data pelatihan dan validasi berasal dari *Stanford Dog Dataset* dari *platform online* pembelajaran mesin Kaggle sebanyak 2.758 data. Sementara data pengujian yang diolah dan dikumpulkan dari berbagai sumber berjumlah 120 data.

2.2 Landasan Teori

2.2.1 Data

Data adalah sebuah kumpulan objek dan atributnya yang menggambarkan suatu kejadian yang terjadi, yang kemudian data tersebut akan diolah dan diterapkan dalam sistem ^[1,2].

2.2.2 Informasi

Informasi adalah hasil pengolahan data yang telah dibentuk atau dimanipulasi sesuai dengan keperluan tertentu bagi penggunaannya sehingga menghasilkan nilai yang lebih bermakna dan bermanfaat^[3,4].

2.2.3 Artificial Intelligence

Kecerdasan buatan adalah salah satu bagian dari ilmu komputer yang mempelajari membuat mesin komputer dapat melakukan seperti yang dilakukan manusia menggunakan bantuan dari kecerdasan dan perilaku manusia, hewan, dan tumbuhan^[5,6].

2.2.4 Computer Vision

Computer vision adalah bidang studi yang mengembangkan teknik untuk menggunakan komputer untuk meniru penglihatan manusia sehingga dapat melihat dan memahami *input* visual^[7,8].

2.2.5 Convolutional Neural Network

CNN adalah arsitektur jaringan syaraf *multilayer feed-forward* yang terinspirasi oleh mekanisme saraf manusia yang terdiri dari beberapa lapisan konvolusi ^[9].

2.2.6 Python

Python adalah bahasa pemrograman tingkat tinggi yang interpretatif dan multiguna, serta dibuat dengan tujuan memudahkan programmer menyelesaikan pekerjaannya. Python juga bersifat *open source* dan sangat kompatibel dengan berbagai *platform*^[10].

2.2.7 Tensorflow

Tensorflow adalah sebuah *framework machine learning* yang bersifat *open-source* dan sangat cocok untuk perhitungan yang melibatkan *array* serta TensorFlow memberikan kemampuan untuk menjalankan kode baik pada GPU maupun CPU. ^[11,12].

2.2.8 Inceptionv3

Inceptionv3 adalah model jaringan saraf dalam yang telah dilatih sebelumnya untuk mengklasifikasikan gambar dan memiliki beberapa lapisan konvolusional dengan ukuran kernel yang berbeda bekerja berdampingan, masing-masing menghasilkan sejumlah *channel output*^[13,14].

2.2.9 Unified Modelling Language (UML)

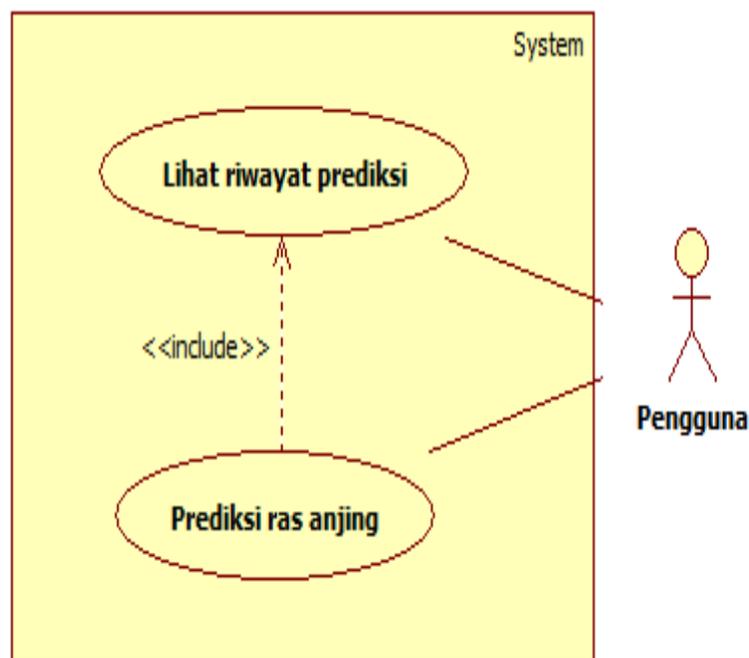
UML adalah standar bahasa untuk mendokumentasikan model dengan mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek menggunakan sintaks dan semantik ^[15,16].

3. HASIL DAN PEMBAHASAN

Aplikasi ini dikembangkan untuk membantu pengklasifikasian ras anjing (*canis domesticus*) melalui prediksi citra grafis. Hasil implementasi metode CNN dengan arsitektur Inceptionv3 yang akan dibahas mencakup:

3.2 Pemodelan Rancangan Aplikasi dengan Diagram UML (Unified Modelling Language)

Dalam merancang sebuah aplikasi diperlukan sebuah gambaran mengenai aplikasi yang akan dibuat. Diagram *use case* digunakan untuk mendeskripsikan interaksi antara pengguna dan admin dengan sistem aplikasi yang dibuat secara berurutan dalam aplikasi yang dirancang.



Gambar 1. Diagram Use Case Aplikasi Prediksi Ras Anjing

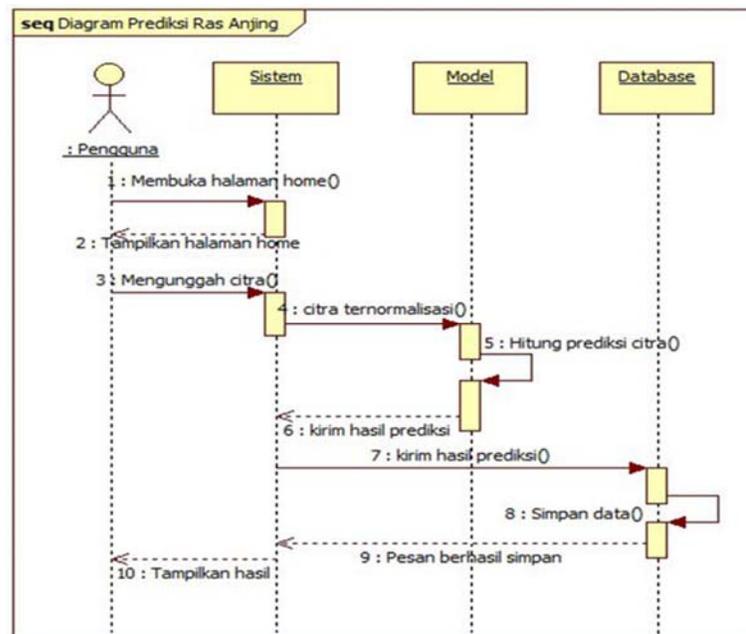
Gambar 1 adalah diagram *use case* untuk aplikasi prediksi ras anjing. Dapat dilihat bahwa terdapat satu aktor yaitu pengguna. Terdapat pula dua *use case* pada sistem yaitu prediksi ras anjing dan lihat riwayat prediksi yang memiliki asosiasi *include*.

Pengguna adalah aktor yang memanfaatkan aplikasi ini untuk memprediksi ras anjing. Pengguna dapat melakukan beberapa fungsi utama dalam aplikasi ini, seperti login, register, memprediksi ras anjing dengan mengunggah gambar anjing, melihat riwayat prediksi yang telah dilakukan oleh akun tersebut sebelumnya, dan logout.

Sequence Diagram atau diagram urutan berfungsi untuk menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Untuk menggambar diagram urutan maka harus mengetahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang menjadi objek itu.

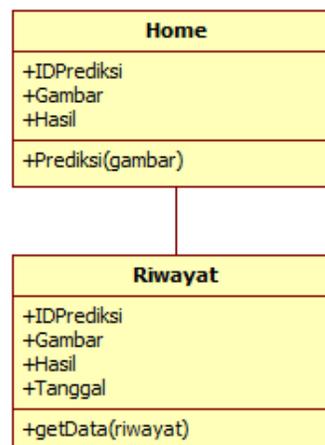
Diagram urutan sangat penting dalam pemodelan perangkat lunak karena membantu memahami bagaimana objek-objek berinteraksi satu sama lain dalam sistem. Diagram urutan digunakan untuk menggambarkan interaksi antara objek-objek yang terlibat dalam sebuah *use case*.

Diagram urutan pada Gambar 2 menggambarkan proses urutan waktu hidup objek dan pesan yang dikirimkan dan diterima antara pengguna dengan sistem pada menu bagian prediksi. Terdapat 1 aktor yaitu pengguna dan 3 objek pada diagram urutan menu prediksi ras anjing, yaitu sistem, model, dan *database*. Pengguna memulai proses prediksi ras anjing dengan membuka form *home* pada aplikasi. Pengguna mengunggah gambar anjing yang ingin diprediksi rasnya. Setelah pengguna mengunggah gambar, gambar tersebut kemudian dinormalisasi dalam form *home*.



Gambar 2. Diagram Urutan Prediksi Ras Anjing

Setelah gambar dinormalisasi, model kemudian menghitung prediksi ras anjing berdasarkan gambar tersebut. Hasil prediksi oleh model kemudian disimpan dalam database agar dapat ditampung sebagai riwayat prediksi. Terakhir, form *home* akan menampilkan hasil prediksi beserta penjelasan singkat dengan persentase 3 kelas tertinggi kepada pengguna.



Gambar 3. Diagram Kelas Aplikasi Prediksi Ras Anjing

Diagram kelas adalah jenis diagram struktur statis dalam UML yang menggambarkan struktur sistem dengan menunjukkan sistem *class*, atributnya, metode, dan hubungan antar objek. Gambar 3 merupakan diagram kelas yang menggambarkan aplikasi prediksi ras anjing yang akan dirancang.

Terdapat dua *class* pada aplikasi yang akan dirancang yaitu *class home* dan *class riwayat*. Pada *class* terdapat beberapa atribut dan metode yang dimiliki. *Class home* memiliki operasi *prediksi(gambar)* dan atribut *idprediksi*, *gambar* dan *hasil*. Sementara itu, *class riwayat* memiliki operasi *getData(riwayat)* dan atribut *idprediksi*, *gambar*, *hasil* serta *tanggal*. Kedua *class* tersebut saling berkaitan dan dihubungkan dengan garis asosiasi.

3.3 Implementasi Algoritma CNN pada Aplikasi

Data yang digunakan pada pembuatan aplikasi ini didapatkan dari situs pembelajaran mesin Kaggle yang diunduh dari <http://vision.stanford.edu/aditya86/ImageNetDogs/>. Tabel 1 adalah rincian data yang digunakan dalam pelatihan dan pengujian mulai dari data tersedikit hingga data terbanyak dengan perbandingan 80:20 dengan total 2.758 gambar. Total gambar tiap kelas berkisar antara 149 gambar sampai dengan 218 gambar.

Tabel 1. Tabel Jumlah Data Per Kelas

Nama Kelas	Test Data (80%)	Train Data (20%)	Jumlah Data
Pekinese	119	30	149
Golden Retriever	120	30	150
Border Collie	120	30	150
Chihuahua	122	30	152
Standard Poodle	127	32	159
French Bulldog	127	32	159
Basset Hound	140	35	175
Alaskan Malamute	142	36	178
Siberian Husky	154	38	192
Beagle	156	39	195
Chow Chow	157	39	196
Shih-Tzu	171	43	214
Samoyed	174	44	218
Pomeranian	175	44	219
Maltese	202	50	252
Total Gambar	2206	552	2758

Gambar 4 memberikan contoh gambar yang mewakili setiap ras anjing. Ras-ras tersebut yang akan digunakan dalam proses pelatihan dan pengujian model klasifikasi menggunakan CNN. Terdapat 15 ras anjing yang digunakan dalam gambar, yaitu Chihuahua, Maltese, Shih-Tzu, Pekinese, Golden Retriever, Alaskan Malamute, Siberian Husky, Standard Poodle, French Bulldog, Chow Chow, Samoyed, Basset Hound, Pomeranian, Border Collie, dan Beagle.

Aplikasi ini dikembangkan sebagai perangkat lunak *desktop* menggunakan bahasa Python, dengan memanfaatkan dua *platform* utama untuk pengembangan model dan antarmuka pengguna yaitu:

3.3.1 Google Colab

Platform ini digunakan untuk melatih dan merancang model prediksi ras anjing. Berbagai pustaka Python seperti TensorFlow, Keras, Numpy, Matplotlib, Os, Pickle, OpenCV, Seaborn, dan Sklearn dimanfaatkan untuk pelatihan model, perancangan arsitektur, pengolahan data, visualisasi, dan manipulasi file. Proses pengembangan model meliputi:

3.3.1.1 Model Initialization

Langkah pertama adalah menginisialisasi model Inceptionv3 sebagai model dasar. Model dasar tersebut sudah dilatih sebelumnya dengan *dataset* ImageNet.

3.3.1.2 Preprocessing data

Citra yang digunakan untuk *training* akan diproses agar seragam sebelum masuk ke dalam model. Proses ini meliputi normalisasi ukuran gambar menjadi 256 x 256 piksel dan konversi ke format RGB.

3.3.1.3 Data Augmentation

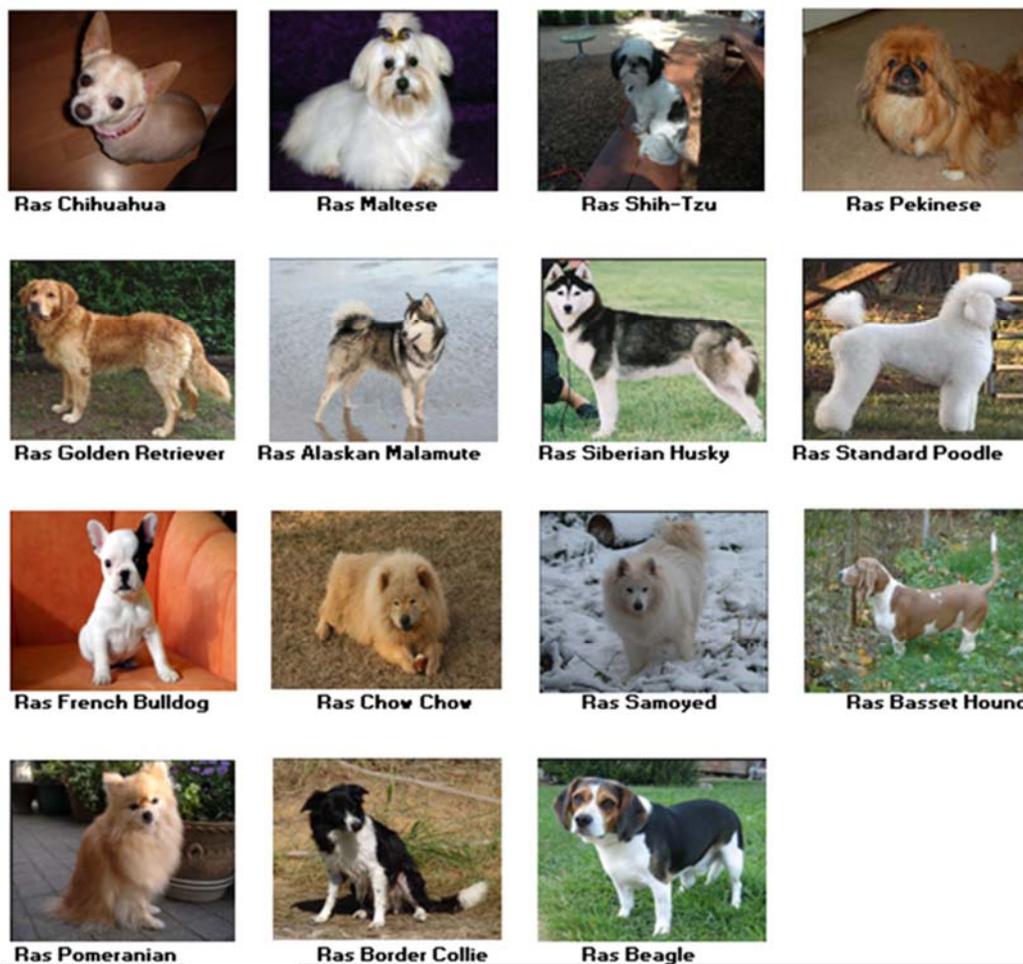
Untuk menghindari resiko *overfitting* saat proses pelatihan, maka dilakukan augmentasi data terhadap data training sehingga variasi data yang dilihat model menjadi lebih banyak.

3.3.1.4 Model Training

Model akan belajar dari *dataset* pelatihan sesuai parameter yang telah ditentukan. Model yang telah dilatih akan digunakan untuk mengklasifikasikan gambar ke dalam kelas ras anjing yang sesuai.

3.3.1.5 Classification

Setelah ekstraksi fitur, aplikasi akan menggunakan model Inceptionv3 untuk mengklasifikasikan gambar ke dalam 3 kelas dengan persentase tertinggi untuk 15 ras anjing terbaik untuk dipelihara di rumah.



Gambar 4. Contoh Data 15 Ras Anjing yang Digunakan

3.3.2 Visual Studio Code

Platform kedua yaitu Visual Studio Code dengan program lingkungan virtual Anaconda untuk melakukan perancangan antarmuka pengguna. Pustaka yang digunakan dalam proses perancangan GUI yakni Tkinter, Os, Pickle, PIL, Numpy, dan lain-lain. Pustaka-pustaka tersebut digunakan dalam proses perancangan antarmuka pengguna, manipulasi data, pemilihan citra, penampilan grafis, dan lain-lain.

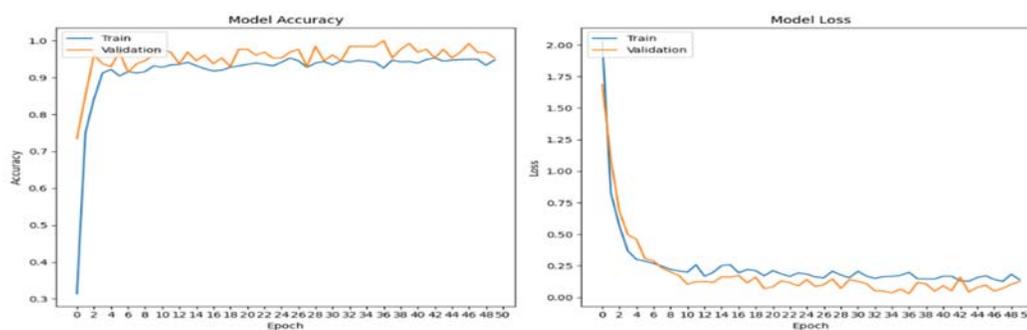
Agar proses perancangan dapat dikustomisasi sebebaskan mungkin, perancangan dilakukan dalam *virtual environment* Anaconda yang telah peneliti buat bernama 'dog' yang telah ditambahkan pustaka-pustaka yang akan digunakan.

3.4 Analisis performa model

Pengujian aplikasi ras anjing ditujukan untuk mengetahui apakah model aplikasi yang dirancang berfungsi dengan baik. Berikut adalah pengujian yang dilakukan dengan menggunakan grafik *accuracy* dan *loss*, *confusion matrix*, serta nilai *precision*, *recall*, dan *F-1 score*.

Gambar 5 adalah grafik *accuracy* dan *loss* yang dihasilkan pelatihan model sampai dengan 50 *epoch*. Grafik sebelah kiri menunjukkan *accuracy* model selama pelatihan. Dapat dilihat bahwa *accuracy* baik untuk data pelatihan (biru) dan validasi (oranye) meningkat seiring berjalannya *epoch*. *Accuracy* pelatihan awal memiliki nilai di bawah 0.4 yang meningkat seiring penambahan *epoch*, sementara *accuracy* validasi dimulai dari antara 0.7 sampai dengan 0.8.

Ada pola penurunan dalam *accuracy* validasi pada *epoch* 6 dan *epoch* 27, namun tren pergerakan garis *accuracy* masih menunjukkan arah *uptrend* serta grafik secara keseluruhan masih mengarah pada konvergensi. Angka *accuracy* pelatihan akhir berhenti di atas angka 0.95, sementara angka *accuracy* pengujian akhir berhenti di antara angka 0.9 dan 0.95. Hal ini menunjukkan walaupun model mengalami ketidakstabilan pada beberapa data validasi, namun model cenderung belajar dengan baik.



Gambar 5. Grafik Accuracy dan Loss Model

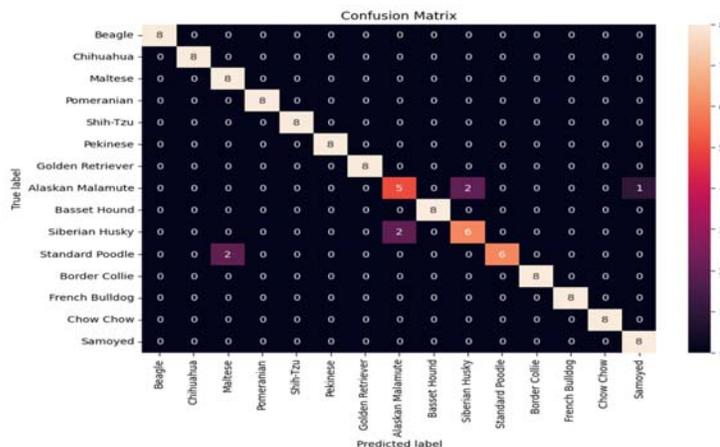
Evaluasi performa model menunjukkan tren positif selama proses pelatihan. Grafik sebelah kanan menunjukkan *loss* model selama pelatihan. Grafik *loss* untuk data pelatihan dan validasi memperlihatkan penurunan konsisten seiring bertambahnya *epoch*. Dapat dilihat bahwa *loss* baik untuk data pelatihan (biru) dan validasi (oranye) menurun seiring bertambahnya *epoch*. *Loss* pelatihan awal memiliki nilai di atas 2 yang menurun seiring penambahan *epoch*, sementara *loss* validasi dimulai dari antara 1.75 sampai dengan 1.5.

Tren pergerakan garis *loss* pelatihan dan validasi masing menunjukkan arah *downtrend* serta grafik secara keseluruhan masih mengarah pada konvergensi. Terjadi pendekatan antara kedua garis *loss* sehingga jarak antar garis semakin kecil. Angka *loss* pelatihan dan validasi akhir masing-masing berhenti di bawah angka 0.25. Pada *epoch* terakhir, nilai *accuracy* dan *val_accuracy* masing-masing adalah 94,73% dan 95,31%. Sementara itu nilai *loss* dan *val_loss* masing-masing adalah 13,89 dan 12,82.

Hal ini merupakan pertanda yang baik, menandakan bahwa model tidak mengalami *overfitting* karena persentase *accuracy* tidak jatuh. Jika kedua garis *loss* tersebut berada dalam jarak yang minimal, itu menandakan bahwa model cenderung memiliki kemampuan yang baik untuk menggeneralisasi dari data latih ke data validasi. Hasil ini menunjukkan potensi model untuk memprediksi data baru (data yang belum pernah dilihat sebelumnya) dengan akurat.

Gambar 6 adalah *confusion matrix* yang dihasilkan dari model dengan arsitektur Inceptionv3 dengan parameter *val_true_labels* dan *val_predictions*. Model dapat mengklasifikasikan beberapa jenis anjing seperti Beagle, Chihuahua, Maltese, Pomeranian, Shih-Tzu, Pekinese, Golden Retriever, Basset Hound, Border Collie, French Bulldog, Chow Chow dan Samoyed dengan akurasi sangat baik. Hal ini dikarenakan *confusion matrix* membentuk pola diagonal utama yang konsisten dengan prediksi benar mayoritas berjumlah 8 (total dari jumlah foto validasi).

Terdapat 3 ras anjing yang mendapat hasil prediksi yang benar kurang dari 8, yaitu Standard Poodle dengan nilai 6, Siberian Husky dengan nilai 6, Alaskan Malamute dengan nilai 5. Dari hasil tersebut juga, dapat diketahui bahwa ras anjing yang paling sulit dikenali dengan model arsitektur Inceptionv3 yang telah dilatih adalah Alaskan Malamute dengan prediksi 5 foto yang benar dari total 8 foto.



Gambar 6. Grafik Confusion Matrix

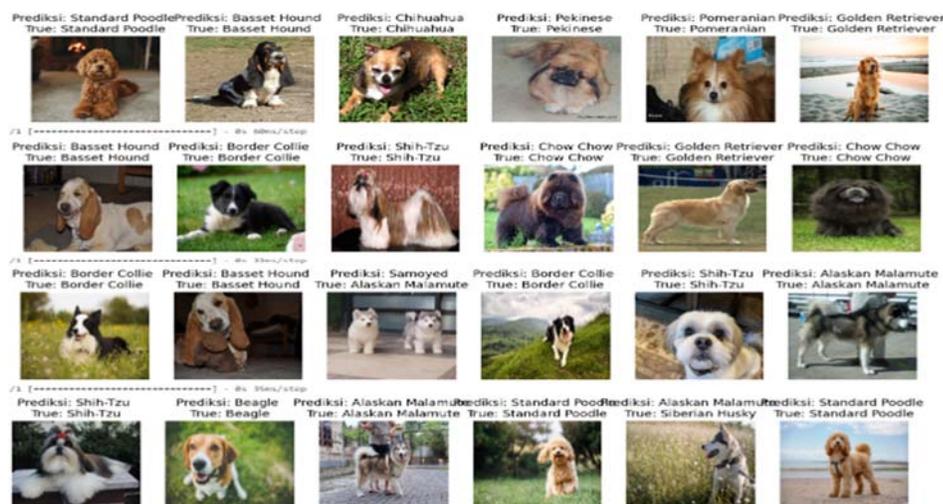
Berdasarkan *Classification Report* yang didapatkan, dapat dilakukan analisis pada performa model pengenalan ras anjing sebagai berikut. Terdapat 11 kelas yang memiliki nilai *precision* 1.00 (100% keberhasilan dalam mengkategorikan foto pengujian ke dalam kelas yang benar), yaitu kelas Beagle, Chihuahua, Pomeranian, Shih-Tzu, Pekinese, Golden Retriever, Basset Hound, Standard Poodle, Border Collie, French Bulldog, dan Chow Chow. Kelas yang tidak memiliki nilai *precision* 1.00 adalah kelas Maltese (0.80), Alaskan Malamute (0.71), Siberian Husky (0.75), dan Samoyed (0.89).

Tabel 2. Tabel Classification Report

Classification Report				
Nama Kelas	Precision	Recall	F1-Score	Support
Beagle	1.00	1.00	1.00	8
Chihuahua	1.00	1.00	1.00	8
Maltese	0.80	1.00	0.89	8
Pomeranian	1.00	1.00	1.00	8
Shih-Tzu	1.00	1.00	1.00	8
Pekinese	1.00	1.00	1.00	8
Golden Retriever	1.00	1.00	1.00	8
Alaskan Malamute	0.71	0.62	0.67	8
Basset Hound	1.00	1.00	1.00	8
Siberian Husky	0.75	0.75	0.75	8
Standard Poodle	1.00	0.75	0.86	8
Border Collie	1.00	1.00	1.00	8
French Bulldog	1.00	1.00	1.00	8
Chow Chow	1.00	1.00	1.00	8
Samoyed	0.89	1.00	0.94	8
Accuracy			0.94	120
Macro avg	0.94	0.94	0.94	120
Weighted avg	0.94	0.94	0.94	120

Terdapat 12 kelas yang memiliki nilai *recall* 1.00 (100% keberhasilan dalam mengkategorikan foto pengujian positif dibanding jumlah total sampel positif, yaitu kelas Beagle, Chihuahua, Pomeranian, Shih-Tzu, Pekinese, Golden Retriever, Basset Hound, Maltese, Border Collie, Samoyed, dan Chow Chow.

Kelas yang tidak memiliki nilai *recall* 1.00 adalah Alaskan Malamute (0.67), Siberian Husky (0.75), dan Standard Poodle (0.75). Nilai *recall* yang rendah untuk beberapa kelas mengindikasikan model masih kesulitan dalam mendeteksi semua sampel yang seharusnya termasuk pada kelas Alaskan Malamute, Siberian Husky, dan Standard Poodle. Hampir semua kelas memiliki nilai *F-1 Score* 1.00, kecuali Maltese (0.89), Alaskan Malamute (0.67), Siberian Husky (0.75), dan Standard Poodle (0.86). Ini menunjukkan bahwa model memiliki performa yang sangat baik dalam mengklasifikasikan sebagian besar kelas.



Gambar 7. Contoh Visualisasi Hasil Prediksi

Semua kelas memiliki nilai *support* 8, yang berarti terdapat 8 sampel untuk setiap kelas dalam *dataset* pengujian. Nilai *support* yang sama untuk semua kelas anjing menunjukkan jumlah gambar yang digunakan untuk evaluasi model sama. Dengan berdasarkan grafik akurasi dan kerugian serta laporan klasifikasi, model memiliki performa klasifikasi yang baik, dengan *precision*, *recall*, dan *F-1 score* yang tinggi untuk sebagian besar kelas.

Namun, ada beberapa kelas seperti Maltese, Alaskan Malamute, Siberian Husky, dan Standard Poodle yang memiliki performa sedikit lebih rendah. Dapat disimpulkan pula bahwa kelas tersulit untuk dideteksi oleh model adalah Alaskan Malamute dengan *precision* 0.71, *recall* 0.67, dan *F-1 score* 0.67, disusul oleh Siberian Husky dengan *precision* 0.75, *recall* 0.75, dan *F-1 score* 0.75, kemudian Standard Poodle dengan *precision* 1.00, *recall* 0.75, dan *F-1 score* 0.86.

3.5 Tampilan Antarmuka Pengguna Aplikasi Prediksi Ras Anjing

Perancangan tampilan antarmuka (*user interface*) adalah sebuah rancangan gambaran mengenai aplikasi yang akan dirancang dan perancangan dibuat untuk mempermudah atau membantu pengguna agar mudah mengakses, memahami dan menggunakan antarmuka aplikasi dengan nyaman. Berikut adalah perancangan antarmuka pengguna pada aplikasi prediksi ras anjing.

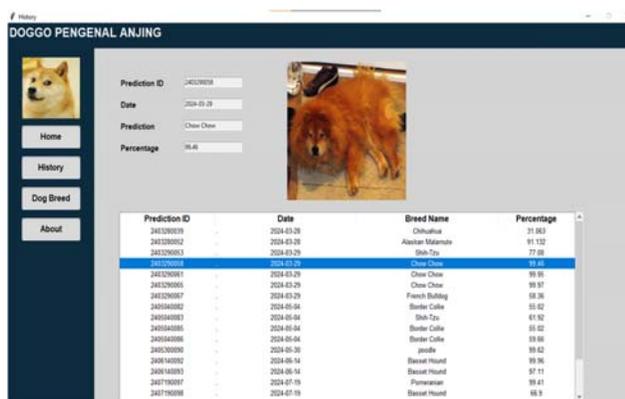
Tampilan pada Gambar 8 menggambarkan antarmuka halaman utama aplikasi. Halaman ini dirancang dengan berbagai elemen yang memiliki fungsi spesifik. Kolom *Insert Picture Here* disediakan untuk memberikan petunjuk kepada pengguna untuk memasukkan gambar pada kotak panel tengah. Terdapat 4 tombol pada menu utama yaitu *camera*, *predict*, *choose picture*, dan hapus.

Tombol *camera* berfungsi untuk membuka kamera pada perangkat pengguna. Tombol *predict* berfungsi untuk menampilkan prediksi 3 ras anjing tertinggi setelah gambar dimasukkan. Sementara itu, tombol *choose picture* digunakan untuk memilih gambar yang terdapat pada penyimpanan perangkat pengguna.



Gambar 8. Tampilan Halaman Home

Gambar 9 menunjukkan tampilan halaman riwayat. Halaman ini berfungsi untuk menampilkan riwayat prediksi yang sudah dilakukan kepada pengguna. Halaman ini didesain dengan tujuan spesifik untuk menyajikan informasi mengenai seluruh aktivitas prediksi yang telah dilakukan oleh pengguna. Fungsi utamanya adalah memberikan akses mudah dan terorganisir terhadap data historis, memungkinkan pengguna untuk meninjau, menganalisis, dan membandingkan hasil-hasil prediksi sebelumnya dengan efisien.



Gambar 9. Tampilan Halaman Riwayat

Struktur halaman riwayat dirancang dengan mempertimbangkan aspek ergonomi dan konsistensi desain. Di sisi kiri halaman, terdapat panel navigasi yang mencakup opsi-opsi seperti *Home*, *History*, *Dog Breed*, dan *About*. Penempatan panel navigasi ini secara strategis mempertahankan keseragaman antarmuka dengan halaman-halaman lainnya dalam aplikasi, memudahkan pengguna untuk beralih antar fitur dengan lancar.

Komponen utama halaman riwayat adalah tabel informasi yang menampilkan data prediksi secara terstruktur. Setiap entri dalam tabel ini mencakup beberapa elemen yaitu ID Prediksi, tanggal prediksi, hasil prediksi ras anjing, dan persentase. Untuk meningkatkan pengalaman pengguna dan fungsionalitas, tabel riwayat dilengkapi dengan fitur *scroll* dan *sort*.

4. KESIMPULAN

Berdasarkan hasil perancangan dan pengujian aplikasi prediksi ras anjing, kesimpulan yang dapat diambil adalah sebagai berikut:

- a. Dari hasil pengujian dalam penelitian ini menunjukkan bahwa pengimplementasian metode CNN dengan arsitektur Inceptionv3 ke dalam aplikasi prediksi ras anjing dapat bekerja dengan baik dan lancar. Akurasi prediksi ras anjing yang dilakukan model adalah 94,17%, *recall* 94,35%, *precision* 94,16%, dan *F-1 score* 94,02%.
- b. Model belajar dengan baik selama 50 *epoch* dan akurasi bertambah secara konsisten. Grafik akurasi dan kerugian pada data latih dan data validasi masing-masing menunjukkan konvergensi sehingga performa model dapat dikategorikan baik.
- c. Aplikasi prediksi ras anjing yang dirancang dapat berfungsi dengan baik. Aplikasi ini dapat membantu pengguna untuk melakukan prediksi ras anjing melalui pemilihan gambar pada direktori maupun melalui kamera perangkat.
- d. Dari analisis pengujian, hasil prediksi aplikasi memiliki tingkat akurasi yang tinggi, namun hasil prediksi yang dilakukan model belum tentu selalu akurat.

5. SARAN

Berdasarkan hasil perancangan dan pengujian aplikasi prediksi ras anjing, masih terdapat kekurangan dalam aplikasi ini sehingga peneliti memberikan saran yang dapat digunakan untuk dikembangkan pada penelitian selanjutnya. Berikut saran yang diberikan untuk penelitian ini:

- a. Dalam pengembangan selanjutnya, model dapat ditingkatkan kinerjanya dengan meningkatkan variasi data pelatihan yang mencakup variasi postur, umur sampel anjing, sudut pengambilan gambar, dan kondisi pencahayaan yang berbeda-beda, dan lain-lain.
- b. Pada pengembangan selanjutnya diharapkan dilakukan pelatihan ulang model dengan menambahkan kelas baru pada *dataset* sehingga model dapat mengklasifikasikan lebih banyak kelas anjing.

UCAPAN TERIMA KASIH

Peneliti menyampaikan apresiasi setinggi-tingginya kepada seluruh pihak yang telah berkontribusi dalam memberikan bimbingan, arahan, data, dan saran selama proses penyusunan jurnal penelitian ini. Ucapan terima kasih disampaikan kepada segenap staf pengajar dan dosen di lingkungan Fakultas Teknologi Informasi Universitas Widya Dharma Pontianak atas bimbingan, petunjuk, dan masukan berharga yang diberikan dalam rangka menyelesaikan jurnal ini. Rasa terima kasih juga peneliti sampaikan kepada orangtua dan teman-teman atas dukungan moril dan semangat yang diberikan. Tidak lupa peneliti mengucapkan terima kasih kepada semua pihak yang turut membantu namun tidak dapat disebutkan satu per satu.

DAFTAR PUSTAKA

- [1] Werdiningsih, I., Nuqoba, B., & Muhammadun. (2020). *Data Mining Menggunakan Android, Weka, dan SPSS*. Surabaya: Airlangga University Presss.
- [2] Kristanto, A. (2018). *Perancangan Sistem Informasi dan Aplikasinya (Edisi Revisi)*. Yogyakarta: Gava Media.
- [3] Prehanto, D. R. (2020). *Buku Ajar Konsep Sistem informasi*. Surabaya: Scopindo Media Pustaka.
- [4] Prabowo, M. (2020). *Metodologi Pengembangan Sistem Informasi*. Salatiga: LP2M IAIN Salatiga.

- [5] Susatyono, J. D. (2021). *Kecerdasan Buatan, Kajian Konsep dan Penerapan*. Semarang: Yayasan Prima Agus Teknik.
- [6] Sarno, R., Sabilla, S. I., Malikhah, Purbawa, D. P., & Ardani, M. H. (2022). *Machine Learning dan Deep Learning-Konsep dan Pemrograman Python*. Yogyakarta: Andi Offset.
- [7] Gonzalez, R. C., & Woods, R. R. (2018). *Digital Image Processing Fourth Edition*. New York: Pearson Education.
- [8] Arnita, Marpaung, F., Fitrahuda Aulia, Suryani, N., & Nabila, R. C. (2022). *Computer Vision dan Pengolahan Citra Digital*. Surabaya: Pustaka Aksara.
- [9] Raharjo, B. (2022). *Deep Learning dengan Python*. Semarang: Yayasan Prima Agus Teknik.
- [10] Daqiqil, I. (2021). *Machine Learning: Teori, Studi Kasus dan Implementasi Menggunakan Python*. Riau: UR PRESS.
- [11] Smith, B. (2020). *Deep Learning With Python: Simple and Effective Tips and Tricks to Learn Deep Learning with Python*. US: Amazon Digital Services LLC - KDP Print US.
- [12] Mattmann, C. (2020). *Machine Learning With Tensorflow Second Edition*. New York: Manning Publications Co.
- [13] Sukanto, R. A., & Shalahuddin, M. (2018). *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek Edisi Revisi*. Bandung: INFORMATIKA Bandung.
- [14] Migunani. (2022). *Rekayasa Perangkat Lunak*. Semarang: Yayasan Prima Agus Teknik.