

Klasifikasi Ras Kucing Dengan Pendekatan Convolutional Neural Networks Menggunakan Arsitektur Inception V4

Adryan Putra Pratama^{1*}

Jumadi²

Eva Nurlatifah³

^{1,2,3} Teknik Informatika, Fakultas Sains dan Teknologi, UIN Sunan Gunung Djati Bandung
Jl. AH. Nasution No.105, Cipadung Wetan, Kec. Cibatu, Kota Bandung, Jawa Barat 40614,
Indonesia

¹adry.pprtm@gmail.com, ²jumadi@uinsgd.ac.id, ³evanurlatifah@uinsgd.ac.id

*Penulis Korespondensi:

Adryan Putra Pratama

adry.pprtm@gmail.com

Abstrak

Klasifikasi ras kucing berdasarkan gambar menjadi tantangan karena perbedaan penampilan antar ras yang halus dan pengaruh lingkungan. Penelitian ini mengembangkan sistem klasifikasi otomatis berbasis arsitektur Inception V4 dengan pendekatan CRISP-DM, meliputi pemahaman bisnis, persiapan data, pemodelan, evaluasi, dan penerapan. Dataset yang digunakan berasal dari Oxford IIIT Pet Dataset, mencakup 12 ras kucing populer, dan diproses melalui pembersihan, augmentasi, normalisasi, serta pembagian menjadi data pelatihan (80%) dan validasi (20%). Model dilatih selama 25 epoch, mencapai akurasi validasi tertinggi 93.31%, dengan rata-rata precision, recall, dan f1-score sebesar 93%. Sistem ini diterapkan dalam aplikasi web berbasis Flask, memungkinkan klasifikasi real-time melalui unggahan gambar. Meskipun performa keseluruhan sangat baik, ras tertentu seperti Bengal menunjukkan peluang untuk perbaikan. Hasil ini menunjukkan bahwa model memiliki potensi besar untuk mendukung diagnosis kesehatan hewan peliharaan dan pelestarian ras. Penelitian ini memberikan kontribusi signifikan dalam pengembangan teknologi klasifikasi berbasis gambar, dengan saran untuk meningkatkan performa melalui augmentasi berbasis GAN dan pengujian pada dataset yang lebih beragam untuk generalisasi yang lebih baik.

Kata Kunci: Ras kucing; CNN; Inception V4; Pengenalan Gambar; Klasifikasi

Abstract

Classifying cat breeds based on images presents challenges due to subtle differences in appearance among breeds and environmental influences. This study developed an automated classification system utilizing the Inception V4 architecture with a CRISP-DM approach, encompassing business understanding, data preparation, modeling, evaluation, and deployment. The dataset used was derived from the Oxford IIIT Pet Dataset, covering 12 popular cat breeds, and underwent cleaning, augmentation, normalization, and partitioning into training (80%) and validation (20%) datasets. The model was trained over 25 epochs, achieving a highest validation accuracy of 93.31% with average precision, recall, and f1-score of 93%. The system was implemented as a Flask-based web application, enabling real-time classification through image uploads. While overall performance was strong, certain breeds such as Bengal exhibited potential for further improvement. The findings demonstrate the model's significant potential to support pet health diagnosis and breed conservation efforts. This study contributes substantially to the development of image-based classification technology, with recommendations for performance enhancements through GAN-based data augmentation and testing on more diverse datasets to improve generalizability.

Keywords: Cat race; CNN; Inception V4; Image Recognition; Classification

1. Pendahuluan

Kucing, atau Felis catus [1], adalah mamalia karnivora yang populer sebagai hewan peliharaan di seluruh dunia. Sejarahnya sebagai pemburu hama di pemukiman manusia kuno telah bertransformasi menjadi peran mereka sebagai sahabat dalam rumah tangga modern. Di Indonesia, jumlah pemelihara kucing meningkat signifikan, dari 2,15 juta ekor pada tahun 2016 menjadi 4,80 juta ekor pada tahun 2022 [2], menjadikannya hewan peliharaan terbanyak di

negara ini. Kucing memiliki beragam ras dengan ciri fisik dan perilaku yang berbeda. Menurut Cat Fanciers' Association (CFA), terdapat 41 jenis ras kucing, seperti Persia dan Maine Coon. Namun, mengidentifikasi ras kucing bisa sulit karena variasi dalam penampilan dan faktor lingkungan yang mempengaruhi mereka. Hal ini menimbulkan tantangan bagi pemilik dalam merawat kucing sesuai kebutuhan spesifik setiap ras.

Penelitian ini menggunakan hanya 12 ras kucing dari total 41 ras yang diakui oleh Cat Fanciers' Association. Pemilihan ini dilakukan untuk fokus pada ras-ras yang paling umum dipelihara dan dikenal di Indonesia, serta untuk memudahkan pengumpulan data yang berkualitas. Dengan membatasi jumlah ras, penelitian dapat lebih mendalam dalam analisis dan pengembangan model klasifikasi yang lebih akurat. Selain itu, pemilihan 12 ras ini memungkinkan peneliti untuk mengelola dataset dengan lebih efisien, mengingat tantangan dalam mengumpulkan gambar berkualitas tinggi untuk setiap ras yang ada. Dengan demikian, meskipun jumlah ras yang dianalisis lebih sedikit, penelitian ini tetap dapat memberikan kontribusi signifikan dalam pengembangan sistem klasifikasi ras kucing.

Mengidentifikasi ras kucing bisa sulit karena variasi dalam penampilan dan faktor lingkungan yang mempengaruhi mereka. Hal ini menimbulkan tantangan bagi pemilik dalam merawat kucing sesuai kebutuhan spesifik setiap ras. Permasalahan ini semakin kompleks karena kurangnya pengetahuan pemilik tentang karakteristik masing-masing ras, yang dapat menyebabkan kesalahan dalam perawatan dan pemeliharaan. Selain itu, ketidakpastian dalam mengenali ras kucing dapat mengakibatkan kesulitan dalam memberikan nutrisi yang tepat dan perawatan kesehatan yang sesuai, sehingga mempengaruhi kualitas hidup kucing tersebut.

Terdapat beberapa penelitian terdahulu yang memiliki keterkaitan dengan penelitian yang diusulkan, atas dasar tersebut lah dilakukan kajian literatur pada penelitian ini untuk mengidentifikasi perbedaan pembahasan yang dihadirkan. Terdapat penelitian serupa yang bertujuan untuk mampu mengenali dan mengklasifikasikan jenis ras kucing dengan menggunakan YOLOv5, namun mendapatkan hasil terbaik pada epoch ke 60 dan batch size diberikan nilai 16 [3]. Penelitian terdahulu menunjukkan berbagai hasil yang dapat ditingkatkan dalam pengembangan model klasifikasi ras kucing menemukan bahwa model Xception yang dikombinasikan dengan transfer learning dan fine-tuning menghasilkan akurasi tertinggi sebesar 93.75%, namun masih ada potensi untuk meningkatkan nilai precision, recall, dan f1-score yang masing-masing berada di angka 93.74%, 93.56%, dan 93.64% [4]. Gunawan et al. (2024) mengembangkan model CNN untuk mengenali lima jenis ras kucing, tetapi hanya mencapai akurasi 77.62% dengan akurasi validasi sebesar 60% dengan iterasi 500 kali, menunjukkan adanya ruang untuk perbaikan dalam pengoptimalan model [5].

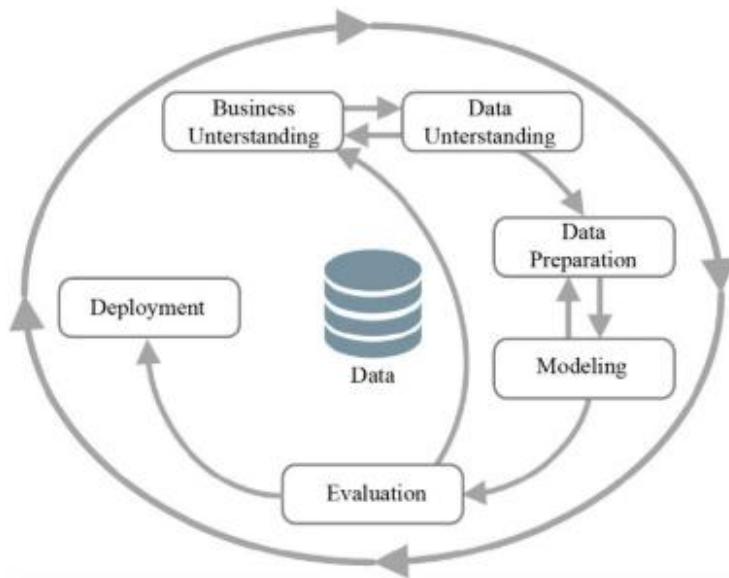
Arsitektur EfficientNet-B0 dipergunakan dan mencapai akurasi tertinggi 98%, namun mengalami masalah overfitting, dengan akurasi optimalnya turun menjadi 95% dan akurasi validasi 91% [6]. Ramadhan & Setiawan (2023) menemukan bahwa MobileNetV2 adalah model dasar terbaik dengan akurasi 82% pada skenario ID-12, yang menunjukkan bahwa ada potensi untuk meningkatkan performa dengan eksplorasi lebih lanjut terhadap arsitektur lain [7]. Dilaporkan juga peningkatan akurasi dari 78% pada epoch pertama menjadi 90% pada epoch ke-20, tetapi stabilitas hasil pada epoch terakhir menunjukkan perlunya lebih banyak iterasi untuk mencapai konsistensi yang lebih baik [8]. Hasil-hasil ini menyoroti pentingnya penelitian lebih lanjut untuk meningkatkan akurasi dan mengatasi masalah overfitting dalam klasifikasi ras kucing.

Pada beberapa penelitian lainnya [9][10] mengusulkan pendekatan Inception V4 untuk mendeteksi COVID-19 secara otomatis melalui citra. Terdapat salah satu model ini mencapai akurasi 99,63% [11], sensitivitas 98,55%, dan spesifisitas 100%, yang secara signifikan lebih unggul dibandingkan metode lain seperti ResNet18 (sensitivitas 94,5%, spesifisitas 92,5%) dan DenseNet121 (sensitivitas 94,3%, spesifisitas 86,0%) yang diuji pada dataset serupa. Tingginya

akurasi dan metrik evaluasi lainnya menunjukkan potensi Inception V4 dalam mendukung pengembangan teknologi diagnosis otomatis dalam dunia penelitian.

2. Metode Penelitian

Metode penelitian ini akan diuraikan menggunakan kerangka CRISP-DM [12][13][14] untuk pengembangan sistem klasifikasi ras kucing berbasis gambar dengan arsitektur Inception V4.



Gambar 1. Proses CRISP-DM [15]/[16]

Business Understanding

Tujuan penelitian ditetapkan untuk mengembangkan sistem klasifikasi ras kucing yang akurat berdasarkan gambar, serta menentukan jenis data mining yang digunakan yaitu klasifikasi, dengan kriteria keberhasilan yang mencakup akurasi dan presisi model. Selanjutnya, pada fase Data Understanding, data dikumpulkan dari dataset publik bernama Oxford IIIT [17] dan hanya mengambil ras kucing dari dataset tersebut yang mencakup 12 kelas ras kucing yang terdiri dari Abyssinian, Bengal, Birman, Bombay, British, Shorthair, Egyptian Mau, Maine Coon, Persian, Ragdoll, Russian Blue, Siamese, Sphynx. Analisis awal dilakukan untuk memahami kualitas data dan distribusi ras dalam dataset.

Data Preparation

Pemilihan data dilakukan dengan menentukan kriteria inklusi dan eksklusi untuk memilih gambar yang akan digunakan dalam pelatihan dan pengujian model. Kriteria inklusi mencakup gambar kucing yang jelas, terlabeli dengan benar, dan mencakup berbagai ras yang ingin dipelajari, dalam hal ini mencakup 12 ras yaitu Abyssinian, Bengal, Birman, Bombay, British, Shorthair, Egyptian Mau, Maine Coon, Persian, Ragdoll, Russian Blue, Siamese, Sphynx. Gambar-gambar ini harus memiliki resolusi yang cukup tinggi untuk memungkinkan ekstraksi fitur yang akurat. Di sisi lain, kriteria eksklusi meliputi gambar yang buram, tidak relevan, atau tidak terlabeli dengan benar, serta gambar kucing yang tidak sesuai dengan ras yang ditentukan. Selain itu, gambar dengan pencahayaan yang buruk atau latar belakang yang terlalu rumit juga dikeluarkan dari dataset untuk menghindari kebingungan dalam proses klasifikasi. Dengan menerapkan kriteria ini, diharapkan model dapat dilatih dengan data yang berkualitas tinggi dan representatif. Pra-pemrosesan data mencakup pembersihan data untuk menangani kualitas data yang buruk, serta melakukan augmentasi data, normalisasi, dan pembagian dataset menjadi set pelatihan dan pengujian yang dimana dibagi menjadi 80% data untuk pelatihan dan 20% untuk validasi.

Modeling

Arsitektur Inception V4 dari CNN dipilih sebagai model klasifikasi. Model dibangun dengan mengatur parameter pelatihan dan melakukan tuning hyperparameter awal, kemudian dilatih menggunakan dataset pelatihan dan diuji pada subset dataset pengujian untuk mengevaluasi performa awal. Arsitektur Inception V4 ini adalah jaringan saraf konvolusi yang dirancang untuk tugas klasifikasi gambar. Prosesnya dimulai dari lapisan input yang menerima gambar berukuran 299x299 piksel dengan 3 saluran warna (RGB). Setelah itu, gambar diproses oleh lapisan "Stem," yang melakukan serangkaian konvolusi untuk menghasilkan peta fitur berukuran 35x35 dengan 384 saluran [18][19]. Lapisan ini menjadi fondasi dasar yang menangkap pola-pola awal dari gambar input. Selanjutnya, terdapat empat modul "Inception-A" yang berfungsi menangkap berbagai tingkat fitur spasial dengan keluaran yang tetap berukuran 35x35x384. Setelah modul ini, lapisan "Reduction-A" digunakan untuk mengurangi dimensi spasial dari 35x35 menjadi 17x17 sambil meningkatkan kedalaman fitur menjadi 1024. Proses ini membantu memampatkan peta fitur tanpa kehilangan informasi penting. Kemudian, tujuh modul "Inception-B" ditambahkan untuk mengekstraksi fitur yang lebih kompleks sambil mempertahankan ukuran keluaran 17x17x1024. Setelah itu, lapisan "Reduction-B" kembali mengurangi dimensi spasial dari 17x17 menjadi 8x8 dengan kedalaman yang meningkat menjadi 1536. Reduksi ini penting untuk persiapan tahap ekstraksi fitur yang lebih mendalam. Tiga modul "Inception-C" selanjutnya difokuskan untuk menangkap pola tingkat tinggi, dengan keluaran berukuran 8x8x1536. Setelah modul ini, diterapkan lapisan average pooling global, yang mereduksi dimensi spasial menjadi vektor berdimensi 1536 dengan mengambil rata-rata dari setiap peta fitur [20]. Langkah ini memadatkan informasi menjadi satu vektor saja. Kemudian, lapisan dropout dengan keep rate 0,8 (atau 20% dropout) digunakan untuk mengurangi overfitting pada model. Akhirnya, lapisan softmax menghasilkan probabilitas untuk 1000 kelas, yang digunakan untuk menentukan keputusan klasifikasi akhir [21]. Setiap lapisan dan modul dalam arsitektur Inception V4 dirancang untuk menangkap fitur yang semakin rinci dan abstrak dari gambar input, menjadikannya efektif dalam tugas klasifikasi gambar dengan tingkat akurasi yang tinggi. Perhitungan dalam Inception V4 melibatkan hasil konvolusi antara input dan filter, serta penerapan normalisasi batch untuk menormalkan data sebelum memasukkannya ke dalam fungsi aktivasi. Struktur lapisan dalam Inception V4 mencakup berbagai jenis lapisan seperti ConvNormAct, MaxPool, AvgPool, dan Inception Modules. Dengan desain ini, Inception V4 mampu menangkap berbagai fitur dari gambar secara efisien, menjadikannya salah satu arsitektur CNN yang paling kuat untuk tugas klasifikasi gambar.

Evaluation

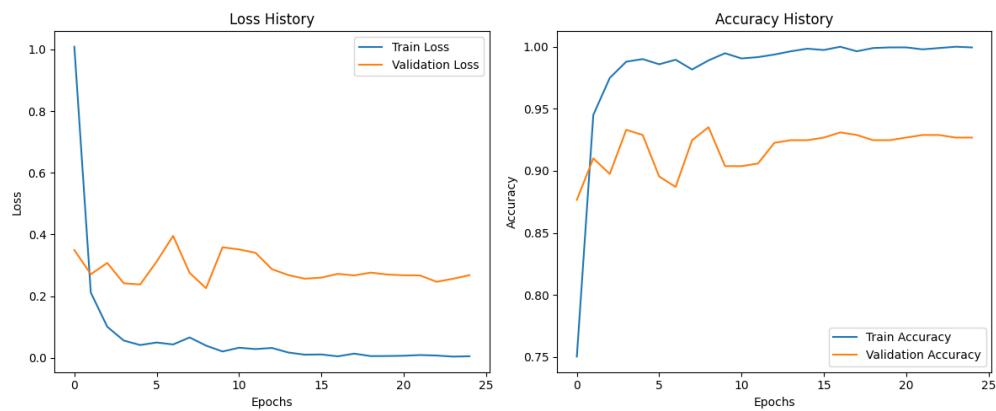
Fase Evaluation dilakukan untuk memeriksa hasil klasifikasi terhadap tujuan bisnis yang ditetapkan. Hasil dievaluasi menggunakan metrik seperti akurasi, presisi, recall, dan F1-score. Selain itu, analisis efisiensi juga dilakukan untuk memastikan efisiensi proses model. Umpan balik dari pemangku kepentingan juga dikumpulkan untuk menilai performa, efisiensi, dan kegunaan model, memastikan bahwa hasil yang diperoleh sejalan dengan kebutuhan dan harapan bisnis.

Deployment

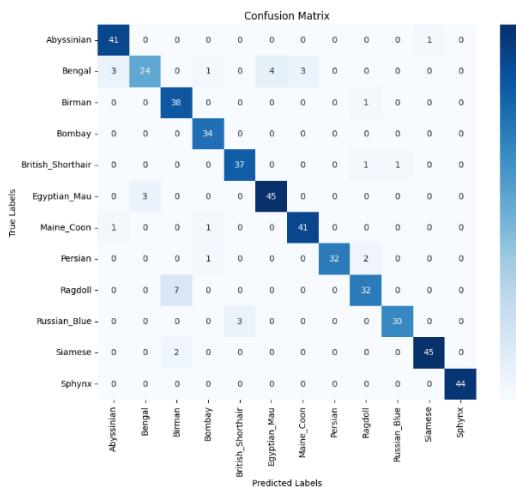
Model klasifikasi diimplementasikan ke dalam aplikasi praktis berbasis web sederhana yang memungkinkan pengguna mengunggah gambar kucing untuk klasifikasi ras secara real-time. Pemantauan performa sistem pasca-implementasi direncanakan serta pemeliharaan dan pembaruan model dilakukan berdasarkan umpan balik dari pengguna. Dengan mengikuti langkah-langkah dalam kerangka CRISP-DM ini, penelitian diharapkan dapat menghasilkan sistem klasifikasi ras kucing yang efektif dan memberikan kontribusi signifikan dalam pengembangan teknologi pengenalan objek berbasis gambar.

3. Hasil

Hasil pelatihan model menunjukkan peningkatan kinerja yang signifikan pada data pelatihan dan validasi selama 25 epoch. Waktu pelatihan untuk setiap epoch rata-rata berada di kisaran 62-64 detik, sedangkan waktu validasi berkisar antara 6.47 hingga 7.10 detik. Pada epoch pertama, model memiliki Training Loss sebesar 1.0088 dengan Training Accuracy 75.03%, sedangkan Validation Loss adalah 0.3495 dengan Validation Accuracy 87.66%. Seiring pelatihan, model menunjukkan penurunan konsisten pada Training Loss hingga mencapai nilai serendah 0.0040 di epoch ke-24, dengan Training Accuracy mencapai 100%. Validation Loss mengalami fluktuasi namun secara umum menunjukkan tren penurunan hingga mencapai 0.2259 pada epoch ke-9, dengan Validation Accuracy tertinggi sebesar 93.31%. Namun, pada beberapa epoch terakhir, Validation Loss sedikit meningkat tetapi masih menunjukkan stabilitas pada nilai yang rendah. Validation Accuracy berkisar antara 92.26% hingga 93.31% pada akhir pelatihan, mengindikasikan bahwa model berhasil mempertahankan performa generalisasi yang baik.



Gambar 1. Diagram Perkembangan Akurasi dan Loss Setiap Epochs



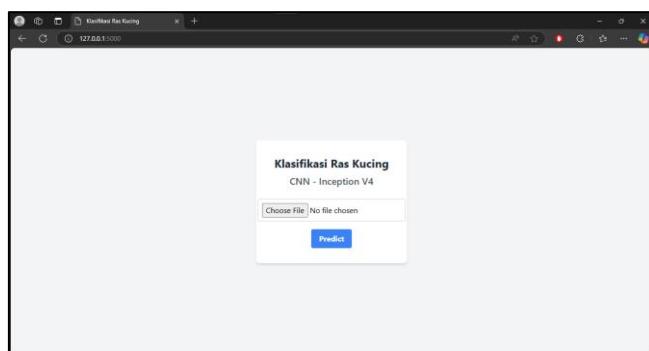
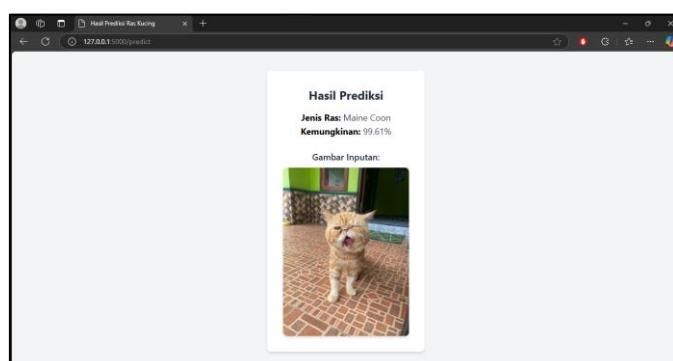
Gambar 2. Hasil Confusion Matrix

Dari hasil laporan klasifikasi yang diperoleh, model menunjukkan performa yang sangat baik dengan rata-rata precision, recall, dan f1-score masing-masing sebesar 93%.

Tabel 1. Classification Report

	Precision	Recall	F1-score	Support
Abyssinian	0.91	0.98	0.94	42
Bengal	0.89	0.69	0.77	35
Birman	0.81	0.97	0.88	39
Bombay	0.92	1.00	0.96	34
British Shorthair	0.93	0.95	0.94	39
Egyptian Mau	0.92	0.94	0.93	48
Maine Coon	0.93	0.95	0.94	43
Persian	1.00	0.91	0.96	35
Ragdoll	0.89	0.82	0.85	39
Russian Blue	0.97	0.91	0.94	33
Siamese	0.98	0.96	0.97	47
Sphynx	1.00	1.00	1.00	44
Accuracy			0.93	478
Macro Avg	0.93	0.93	0.93	478
Weighted Avg	0.93	0.93	0.93	478

Aplikasi website berbasis Flask sederhana untuk mendeteksi ras kucing dirancang sebagai solusi inovatif dalam mengenali jenis kucing berdasarkan gambar yang diunggah pengguna. Dengan menggunakan Python sebagai basis pengembangan, aplikasi ini memanfaatkan framework Flask untuk menyediakan antarmuka yang responsif dan mudah digunakan. Teknologi deteksi berbasis pembelajaran mesin diterapkan melalui model klasifikasi yang telah dilatih menggunakan dataset gambar kucing dari berbagai ras. Pengguna dapat mengunggah gambar kucing mereka, dan aplikasi akan menganalisis serta menampilkan hasil identifikasi ras beserta informasi tambahan terkait karakteristik ras tersebut.

**Gambar 3.** Tampilan Awal Web Sederhana**Gambar 4.** Hasil Prediksi

4. Pembahasan

Hasil penelitian menunjukkan bahwa model berhasil belajar secara efektif dari data pelatihan, sebagaimana terlihat dari penurunan Training Loss yang konsisten hingga mencapai nilai serendah 0.0040 pada epoch ke-24 dan peningkatan Training Accuracy yang mencapai 100%. Tren ini menunjukkan bahwa model mampu meminimalkan error pada data pelatihan secara signifikan. Namun, analisis lebih mendalam terhadap fluktuasi Validation Loss mengindikasikan beberapa poin yang patut dicermati. Validation Loss sempat mengalami kenaikan pada beberapa epoch, khususnya pada epoch ke-6 hingga ke-12, meskipun Validation Accuracy tetap stabil di atas 92%. Hal ini dapat mengindikasikan bahwa model mulai mengalami overfitting ringan pada data pelatihan, di mana model terlalu fokus pada pola spesifik dalam data pelatihan sehingga sedikit mengurangi kemampuannya untuk menggeneralisasi data validasi. Namun, stabilitas Validation Accuracy mengindikasikan bahwa efek overfitting ini relatif terkendali dan tidak berdampak signifikan pada performa generalisasi model.

Dari sisi efisiensi, waktu pelatihan per epoch relatif konsisten, yaitu berkisar antara 62 hingga 64 detik. Ini menunjukkan bahwa pelatihan dilakukan secara efisien tanpa adanya kendala signifikan dalam hal pengelolaan komputasi, seperti bottleneck pada GPU atau CPU yang digunakan. Konsistensi waktu validasi, yang berkisar antara 6.47 hingga 7.10 detik, juga mendukung kesimpulan bahwa proses validasi berjalan stabil dan efisien. Namun, terdapat sedikit peningkatan waktu validasi pada epoch terakhir (7.10 detik), yang dapat diakibatkan oleh ukuran batch data validasi yang sedikit tidak merata pada akhir pelatihan. Analisis ini menunjukkan bahwa pipeline pelatihan model dioptimalkan dengan baik, menghasilkan waktu pelatihan yang stabil dan memungkinkan iterasi cepat untuk mencapai hasil yang diinginkan.

Efisiensi juga dapat dilihat dari kemampuan model mencapai akurasi validasi yang tinggi (maksimal 93.31%) dalam jumlah epoch yang relatif kecil (25 epoch). Model menunjukkan kemampuan untuk mempelajari pola secara cepat, sebagaimana terlihat dari Validation Accuracy yang sudah mencapai 91.00% pada epoch ke-2. Hal ini menunjukkan bahwa arsitektur model dan parameter awal yang digunakan cukup optimal dalam mendukung proses pelatihan. Namun, fluktuasi pada Validation Loss menunjukkan potensi untuk mengimprovisasi efisiensi lebih lanjut dengan tuning hiperparameter seperti learning rate, penggunaan regulasi seperti dropout, atau pendekatan early stopping untuk mencegah overfitting.

Hasil pelatihan menunjukkan performa model yang sangat baik, dengan akurasi validasi maksimum sebesar 93.31%. Namun, terdapat beberapa aspek yang perlu dianalisis lebih lanjut untuk memahami dan meningkatkan hasil model. Performa yang lebih rendah pada beberapa ras kucing dibandingkan ras lainnya dapat disebabkan oleh kemiripan visual antar ras, seperti pola bulu dan warna mata yang hampir identik, yang dapat memperburuk kesalahan prediksi. Kekurangan fitur representasi unik untuk setiap ras juga dapat menjadi salah satu penyebab kurang optimalnya performa pada ras yang lebih kompleks atau sulit dibedakan secara visual. Untuk mengatasi hal ini, augmentasi data menjadi solusi penting, seperti rotasi, flipping, atau perubahan warna gambar untuk meningkatkan variasi data.

Augmentasi data memainkan peran penting dalam mengatasi tantangan dalam pembelajaran, terutama dalam meningkatkan kemampuan model mengenali pola visual yang kompleks. Teknik augmentasi seperti rotasi, perubahan skala, dan penyesuaian warna tidak hanya menambah variasi data tetapi juga membantu model mengenali pola yang lebih kuat tanpa perlu meningkatkan jumlah data secara signifikan. Selain itu, metode lain seperti pemberian bobot lebih tinggi pada ras dengan tingkat kesalahan prediksi yang lebih besar juga dapat membantu memperbaiki performa model secara keseluruhan.

Untuk percobaan dilakukan deployment berbasis aplikasi web sederhana menggunakan Flask untuk melakukan klasifikasi gambar menggunakan model deep learning berbasis PyTorch.

Pertama, aplikasi Flask diinisialisasi, dan folder *UPLOAD_FOLDER* ditentukan sebagai *static/uploads* untuk menyimpan gambar yang diunggah pengguna. Flask memastikan folder ini ada dengan menggunakan *os.makedirs()*. Model klasifikasi yang sudah dilatih disimpan dalam file *best_inceptionv4_model_full.pth*, dan dimuat ke dalam memori menggunakan PyTorch dengan *torch.load()*. Model ini berada dalam mode evaluasi (*model.eval()*), sehingga tidak akan diperbarui selama inferensi. Fungsi *process_image* mengatur preprocessing gambar sebelum diklasifikasikan. Fungsi ini mengubah gambar menjadi tensor, menyesuaikan ukurannya menjadi 256x256 piksel, memotongnya ke ukuran 224x224, mengubah gambar ke bentuk tensor, dan menormalkannya dengan nilai rata-rata dan standar deviasi yang biasa digunakan pada model pre-trained di PyTorch. Aplikasi ini memiliki dua route utama:

1. Route / – Route utama atau home yang akan menampilkan halaman HTML sederhana (*home.html*) dengan formulir untuk mengunggah gambar. Formulir ini memiliki input untuk memilih file gambar dan tombol untuk mengirim gambar ke server.
2. Route /*predict* – Route ini menangani permintaan POST dari formulir. Aplikasi mendapatkan gambar yang diunggah, menyimpannya di folder uploads, dan memprosesnya dengan *process_image*. Tensor gambar kemudian diberikan ke model untuk prediksi. Output model diubah ke probabilitas menggunakan softmax, dan kelas dengan probabilitas tertinggi diambil sebagai prediksi. Prediksi, termasuk nama kelas dan tingkat kepastiannya dalam bentuk persentase, kemudian ditampilkan di halaman *predict.html* bersama dengan gambar yang diunggah.

Terakhir, terdapat fungsi untuk menyajikan gambar yang diunggah melalui route *uploaded_file*, yang memungkinkan Flask untuk melayani file statis dari direktori uploads. Halaman hasil (*predict.html*) menampilkan prediksi kelas kucing, probabilitas prediksi, dan gambar yang diunggah, memberikan antarmuka pengguna yang sederhana dan interaktif untuk klasifikasi gambar kucing.

Hasil ini menunjukkan bahwa proses pelatihan dan validasi model berjalan dengan efisien baik dari sisi waktu maupun performa. Namun, adanya fluktuasi Validation Loss dan sedikit kenaikan waktu validasi pada beberapa epoch terakhir menandakan bahwa masih ada ruang untuk meningkatkan efisiensi dan performa generalisasi model melalui pengaturan hiperparameter yang lebih optimal atau evaluasi lebih lanjut pada distribusi data pelatihan dan validasi. Dari sisi dampak praktis, aplikasi ini memiliki potensi besar dalam konteks yang lebih luas, terutama dalam diagnosis kesehatan hewan peliharaan berbasis ras. Identifikasi ras kucing memungkinkan dokter hewan mengaitkan potensi risiko penyakit genetik tertentu yang umum pada ras tersebut, seperti polycystic kidney disease (PKD) pada kucing Persia atau hypertrophic cardiomyopathy (HCM) pada Maine Coon. Selain itu, aplikasi ini dapat meningkatkan kesesuaian adopsi dengan membantu calon pemilik memahami kebutuhan khusus ras tertentu, serta mempermudah pelaporan kehilangan kucing melalui deskripsi fisik yang lebih akurat. Dalam program pelestarian ras, aplikasi ini juga dapat membantu melacak populasi ras yang langka.

Secara keseluruhan, hasil penelitian ini menunjukkan bahwa model yang dikembangkan telah memiliki performa tinggi dengan efisiensi pelatihan yang baik, meskipun masih terdapat ruang untuk meningkatkan akurasi dan generalisasi model, terutama pada ras tertentu, melalui pengaturan hiperparameter yang lebih optimal dan penambahan fitur representasi yang lebih mendalam. Optimalisasi aplikasi untuk perangkat mobile juga menjadi langkah strategis untuk meningkatkan aksesibilitas dan penerapan teknologi ini secara lebih luas.

Dalam konteks pengembangan aplikasi klasifikasi ras kucing berbasis gambar untuk perangkat mobile, optimalisasi model menjadi aspek penting agar sistem dapat berfungsi efisien di lingkungan dengan sumber daya terbatas. Model Inception V4, meskipun memiliki performa tinggi, dikenal memiliki arsitektur kompleks dan ukuran yang cukup besar, sehingga memerlukan

strategi khusus untuk kompatibilitas pada perangkat dengan keterbatasan memori, daya komputasi, dan konsumsi baterai.

5. Penutup

Penelitian ini berhasil mengembangkan model klasifikasi ras kucing berbasis Inception V4 dengan akurasi validasi tertinggi sebesar 93.31%. Proses pelatihan berjalan efisien, dan model mampu diimplementasikan dalam aplikasi web berbasis Flask untuk klasifikasi real-time. Hasil penelitian ini menunjukkan potensi besar dalam mendukung diagnosis kesehatan hewan berbasis ras, meningkatkan pemahaman pemilik kucing tentang kebutuhan spesifik ras, dan mendukung pelestarian ras yang terancam punah.

Namun, tantangan seperti fluktuasi Validation Loss dan performa rendah pada beberapa ras masih dapat diperbaiki. Penelitian lanjutan disarankan untuk menggunakan augmentasi data berbasis GAN guna meningkatkan variasi data serta menguji model pada dataset yang lebih beragam. Optimalisasi untuk perangkat mobile juga menjadi langkah penting untuk memperluas adopsi teknologi ini. Dengan langkah-langkah tersebut, sistem yang dikembangkan dapat memberikan kontribusi signifikan bagi pengembangan teknologi klasifikasi berbasis gambar dan aplikasinya dalam kehidupan sehari-hari.

6. Referensi

- [1] M. Oedin *et al.*, "Cats Felis catus as a threat to bats worldwide: a review of the evidence," Jul. 01, 2021, *Blackwell Publishing Ltd.* doi: 10.1111/mam.12240.
- [2] P. P. Ridwan, "Ragam Statistik Hewan Peliharaan di Indonesia." Accessed: May 12, 2024. [Online]. Available: <https://goodstats.id/article/ragam-statistik-hewan-peliharaan-di-indonesia-GbtcU>
- [3] M. Ismu Rahayu, Faiqunisa, and Nugraha, "KLASIFIKASI RAS KUCING MENGGUNAKAN METADATA DATASET KAGGLE DENGAN FRAMEWORK YOLO v5," 2023.
- [4] M. A. A. Fawwaz, K. N. Ramadhani, and F. Sthevanie, "Klasifikasi Ras pada Kucing menggunakan Algoritma Convolutional Neural Network(CNN)," 2020.
- [5] R. Gunawan, D. M. I. Hanafie, and A. Elanda, "Klasifikasi Jenis Ras Kucing Dengan Gambar Menggunakan Convolutional Neural Network (CNN)," *Jurnal Interkom: Jurnal Publikasi Ilmiah Bidang Teknologi Informasi dan Komunikasi*, vol. 18, no. 4, pp. 1–8, Jan. 2024, doi: 10.35969/interkom.v18i4.318.
- [6] N. Azahro Choirunisa, T. Karlita, and R. Asmara, "Deteksi Ras Kucing Menggunakan Compound Model Scaling Convolutional Neural Network," *Technomedia Journal*, vol. 6, no. 2, pp. 236–251, Nov. 2021, doi: 10.33050/tmj.v6i2.1704.
- [7] A. T. Ramadhan and A. Setiawan, "Catbreedsnet: An Android Application for Cat Breed Classification Using Convolutional Neural Networks," *Jurnal Online Informatika*, vol. 8, no. 1, pp. 52–60, Jun. 2023, doi: 10.15575/join.v8i1.1007.
- [8] Y. Zhang, J. Gao, and H. Zhou, "Breeds Classification with Deep Convolutional Neural Network," in *ACM International Conference Proceeding Series*, Association for Computing Machinery, Feb. 2020, pp. 145–151. doi: 10.1145/3383972.3383975.
- [9] P. Rachana, A. Kodipalli, and T. Rao, "Comparison Between ResNet 16 and Inception V4 Network for COVID-19 Prediction," 2022, pp. 283–290. doi: 10.1007/978-981-19-5482-5_25.
- [10] A. Sharma, A. Kodipalli, and T. Rao, "Performance of Resnet-16 and Inception-V4 Architecture to Identify Covid-19 from X-Ray Images," in *2022 IEEE 9th Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON)*, 2022, pp. 1–6. doi: 10.1109/UPCON56432.2022.9986372.
- [11] A. Alqahtani *et al.*, "A Transfer Learning Based Approach for COVID-19 Detection Using Inception-v4 Model," *Intelligent Automation and Soft Computing*, vol. 35, no. 2, pp. 1721–1736, 2023, doi: 10.32604/iasc.2023.025597.

- [12] C. Schröer, F. Kruse, and J. M. Gómez, "A systematic literature review on applying CRISP-DM process model," in *Procedia Computer Science*, Elsevier B.V., 2021, pp. 526–534. doi: 10.1016/j.procs.2021.01.199.
- [13] S. Jaggia, A. Kelly, K. Lertwachara, and L. Chen, "Applying the CRISP-DM Framework for Teaching Business Analytics," 2020.
- [14] U. Kannengiesser and J. S. Gero, "MODELLING THE DESIGN OF MODELS: AN EXAMPLE USING CRISP-DM," in *Proceedings of the Design Society*, Cambridge University Press, 2023, pp. 2705–2714. doi: 10.1017/pds.2023.271.
- [15] S. Huber, H. Wiemer, D. Schneider, and S. Ihlenfeldt, "DMME: Data mining methodology for engineering applications - A holistic extension to the CRISP-DM model," in *Procedia CIRP*, Elsevier B.V., 2019, pp. 403–408. doi: 10.1016/j.procir.2019.02.106.
- [16] J. Brzozowska, J. Pizoń, G. Baytikenova, A. Gola, A. Zakimova, and K. Piotrowska, "DATA ENGINEERING IN CRISP-DM PROCESS PRODUCTION DATA – CASE STUDY," *Applied Computer Science*, vol. 19, no. 3, pp. 83–95, 2023, doi: 10.35784/acs-2023-26.
- [17] S. S. D. S. and S. N. Reddy, "Efficient Real-time Breed Classification using YOLOv7 Object Detection Algorithm," in *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, IEEE, Jul. 2023, pp. 1–6. doi: 10.1109/ICCCNT56998.2023.10306688.
- [18] M. S. Nazir *et al.*, "A Novel CNN-Inception-V4-Based Hybrid Approach for Classification of Breast Cancer in Mammogram Images," *Wireless Communications and Mobile Computing*, vol. 2022, 2022, doi: 10.1155/2022/5089078.
- [19] F. Hou, B. Liu, L. Zhuo, Z. Zhuo, and J. Zhang, "Remote Sensing Image Retrieval with Deep Features Encoding of Inception V4 and Largevis Dimensionality Reduction," *Sensing and Imaging*, vol. 22, no. 1, Dec. 2021, doi: 10.1007/s11220-021-00341-7.
- [20] K. M. Karthick Raghunath, V. Vinoth Kumar, M. Venkatesan, K. K. Singh, T. R. Mahesh, and A. Singh, "XGBoost Regression Classifier (XRC) Model for Cyber Attack Detection and Classification Using Inception V4," *Journal of Web Engineering*, vol. 21, no. 4, pp. 1295–1322, May 2022, doi: 10.13052/jwe1540-9589.21413.
- [21] M. A. S. Al Husaini, M. H. Habaebi, T. S. Gunawan, M. R. Islam, E. A. A. Elsheikh, and F. M. Suliman, "Thermal-based early breast cancer detection using inception V3, inception V4 and modified inception MV4," *Neural Computing and Applications*, vol. 34, no. 1, pp. 333–348, Jan. 2022, doi: 10.1007/s00521-021-06372-1.