

Sistem Deteksi Penyakit Kulit Kucing Menggunakan Algoritma *You Only Look Once* (YOLO) v8

Bunga Meilita¹, Wiyli Yustanti²

1,2 Jurusan Teknik Informatika/Program Studi S1 Sistem Informasi, Universitas Negeri Surabaya

1bunga.20037@mhs.unesa.ac.id

2wiyliyustanti@unesa.ac.id

Abstrak— Kucing adalah hewan peliharaan yang populer di Indonesia, dengan jumlah populasi mencapai 4,80 juta ekor pada tahun 2022. Meskipun menggemaskan dan menyenangkan, kucing rentan terkena penyakit, terutama penyakit kulit seperti jamur. Pemilik hewan masih banyak yang kurang memahami gejala penyakit kulit kucing, sehingga penanganan penyakit tidak tepat yang bisa memperparah kondisi kucing. Solusi untuk mengatasi permasalahan tersebut dengan mengimplementasikan algoritma *You Only Look Once* (YOLO) v8 yang dapat dijalankan secara *realtime* untuk mendeteksi penyakit kulit kucing jamur, scabies, lain dan sehat melalui aplikasi android. Berdasarkan hasil uji didapatkan Map score sebesar 0.788, precision sebesar 0.727, recall sebesar 0.769, dan F1-Score sebesar 0.75. Hasil pengujian *white box* berhasil berjalan pada semua test case yang ada. Hasil *blackbox* testing yaitu aplikasi bisa berjalan sesuai yang diharapkan, selain itu hasil uji pada fitur *camera detector* dengan pengujian ditiga jarak yang berbeda didapatkan jarak yang paling optimal untuk melakukan pendeteksian penyakit kulit kucing secara *real time* yaitu 20 cm dengan akurasi pengujian sebesar 0.92. Hasil uji pada fitur *import gambar* menghasilkan keakuratan sebesar 0.92.

Kata Kunci— Kucing, *You Only Look Once* (YOLO), Android, Deteksi, Penyakit Kulit.

I. PENDAHULUAN

Kucing adalah hewan peliharaan yang memiliki rupa menggemaskan. Selain menggemaskan, kucing dapat menjadi teman bermain ataupun mengobrol. Bahkan, tidak sedikit pemelihara kucing ini menganggap kucing sebagai bagian dari anggota keluarga [1]. Pada tahun 2022 jumlah kucing yang dipelihara warga Indonesia mencapai 4,80 juta ekor yang mana jumlah populasi ini naik 2,15 juta ekor dibandingkan pada tahun 2016. Berbeda dengan jumlah populasi anjing sebagai hewan peliharaan yang tercatat hanya 737.400 ekor pada tahun 2022 [2].

Kucing tidak bisa lepas dari penyakit yang akan menyerangnya, oleh karena itu pemilik hendaknya lebih mengawasi kucingnya untuk mencegah hewan peliharaannya terjangkit penyakit maupun virus yang juga bisa menularkan ke pemiliknya. Salah satu penyakit yang sering menjangkit kucing yaitu penyakit kulit, apabila penyakit kulit sudah menyebar hingga melebihi 40% tubuh kucing yang mana berarti kucing tersebut berpotensi mengalami infeksi sekunder yang dapat menyebabkan kematian [3].

Salah satu penyakit kulit kucing yang sering terjadi yaitu jamur. Hal ini berdasarkan hasil observasi di klinik Galaxy Satwa, kasus scabies pada bulan Januari – Oktober 2023 sebanyak 70 kasus, untuk kasus kutu sebanyak 142 kasus, dan

untuk kasus penyakit jamur ada sebanyak 172 kasus. Miripnya gejala penyakit kulit kucing seperti menggaruk dan bulu rontok yang mengakibatkan orang awam kesulitan dalam menentukan penyakit kulit kucing yang diderita kucingnya [4].

Pendeteksian penyakit kulit kucing yang dilakukan dengan menemui dokter hewan, namun menurut hasil wawancara yang dilakukan oleh Widiyanti dkk., dokter hewan di setiap daerah belum tentu ada dan sebagian besar hanya terdapat di kota saja ataupun kesibukan dari pemilik kucing sehingga tidak bisa setiap waktu membawa kucingnya ke dokter hewan. Selain itu, sekitar 75% dari 20 responden memilih untuk merawat dan mengobati sendiri penyakit kulit yang sedang diderita kucing mereka karena tidak semua pemilik kucing mempunyai ekonomi yang cukup untuk datang ke dokter hewan, tentu saja hal ini akan memperburuk kondisi kucing [5].

Berdasarkan survei yang dilakukan oleh Pangestu dkk., sebesar 65% dari responden mengatakan bahwa kucing peliharaannya pernah mengalami penyakit kulit dan sekitar 65% dari 20 responden kesulitan dalam mengenali penyakit kulit kucing yang sedang diderita kucingnya [6]. Oleh karena itu dibutuhkan sistem yang mampu mendeteksi penyakit kulit kucing berdasarkan *image* yang mampu membantu para pemelihara kucing mengidentifikasi atau mengetahui penyakit kulit kucing.

Pada penelitian ini, proses deteksi penyakit kulit kucing menggunakan algoritma *You Only Look Once* (YOLO). YOLO merupakan sebuah algoritma yang dapat melakukan pendeteksian secara *real-time* [7]. YOLO pertama kali dikembangkan oleh Joseph Redmon pada tahun 2015 [8]. Penggunaan metode YOLO menghasilkan akurasi yang mencapai 76%. Hal ini berdasarkan dari hasil penelitian mengenai deteksi jenis penyakit dan hama pada tanaman jagung dengan menggunakan YOLOv5 [9].

Tujuan dari penelitian ini adalah untuk mendapatkan arsitektur terbaik pada algoritma YOLOv8 untuk deteksi penyakit kulit kucing dan mengimplementasikan model YOLOv8 pada aplikasi deteksi penyakit kulit kucing berbasis android. Adapun kelas yang dideteksi dalam penelitian ini yaitu jamur, scabies, lain, dan sehat.

II. METODE PENELITIAN

Pada penelitian ini tahapan-tahapan yang dilakukan dengan menggunakan framework CRIPS-DM, tahapan CRIPS-DM dimulai dari data collection, data preparation, modelling, evaluation, dan deployment. Pada penelitian ini,

Langkah-langkah yang akan dilakukan untuk mendeteksi penyakit kulit kucing sebagai berikut :

A Business Understanding

Berdasarkan latar belakang yang sudah dipaparkan sebelumnya, masalah utama yang berkaitan dengan penelitian ini adalah bagaimana mendeteksi jenis penyakit kulit kucing dari data citra dengan menggunakan *deep learning* YOLOv8, yang nantinya akan menghasilkan tingkat akurasi yang baik.

B Data Understanding

Pada tahapan ini merupakan proses pengumpulan dataset untuk membangun model deteksi penyakit kulit kucing. Dataset penelitian ini, direpresentasikan dengan gambar penyakit ulit kucing jamur, scabies, sehat da penyakit lainnya yang meliputi abses, luka dan tungau telinga. Dalam penelitian ini, data diperoleh dari berbagai sumber seperti Instagram, Facebook, Tik Tok, Telegram, google, Robolow Universe, dan Kaggle dataset.

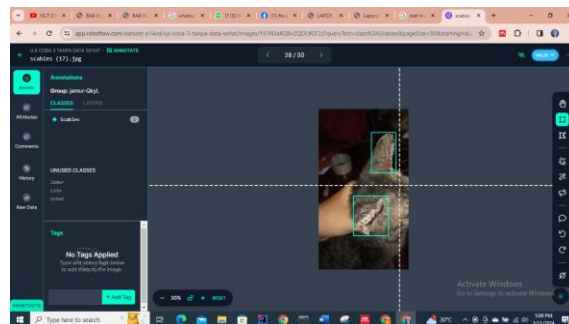
C Data Preparation

Pada tahapan ini dilakukan semua kegiatan guna membangun dataset yang akan dimasukkan ke dalam alat pemodelan. Tugas persiapan data kemungkinan besar akan dilakukan beberapa kali, dan tidak dalam urutan tertentu. Isinya meliputi pemisahan gambar dan data *cleaning*, pelabelan gambar, pembagian gambar untuk training dan testing, dan augmentasi data.

Pada tahapan pemisahan gambar ini data mentah yang sudah didapatkan akan diklasifikasikan secara manual dengan dibantu oleh drh. Muchamad Erfan Efendi . Proses pemisahan gambar dilakukan dengan menamai ulang gambar atau langsung memasukka gambar ke dalam folder sesuai jenis penyakit kulit. Selain pemisahan gambar ,dilakukan juga data *cleaning*. Pada tahapan data *cleaning* ini data mentah akan dibersihkan secara manual agar tidak mengganggu proses deteksi. Data *cleaning* dilakukan bersamaan dengan pemisahan gambar dengan cara menghapus data yang terlalu *blur*, pengambilan gambar terlalu jauh, dan penyakit kulit kucing tidak teridentifikasi jenisnya. Dari proses ini didapatkan total semua data sebanyak 1498 gambar, dimana sebanyak 350 gambar untuk kelas sehat, sebanyak 497 gambar untuk kelas scabies, sebanyak 320 gambar untuk kelas penyakit lain, dan sebanyak 331 gambar untuk kelas jamur.

Setelah melakukan pengumpulan data, selanjutnya melabeli 1498 gambar. Pada proses pelabelan peneliti menggunakan *website* Roboflow, proses pelabelan dilakukan secara manual pada setiap gambar dengan menggunakan Roboflow bisa dilihat pada Gbr.1. Setelah proses pelabelan selesai, setiap gambar akan ada file “.txt”, dari Ketika gambar ada 1498 file “.txt” juka akan berjumlah 1498. Isi dari file “.txt” ini untuk angka pertama merupakan code kelas dimana angka 0 untuk kelas jamur, angka 1 untuk kelas lain, angak 2 untuk kelas Scabies, dan angka 3 untuk kelas sehat. Angka-angka selanjutnya merupakan posisi *bounding box* yaitu titik

koordinat x , koordinat y, tinggi *bounding box*, dan lebar *bounding box*. File “.txt” ini otomatis terbuat Ketika sudah melakukan pelabelan.



Gbr. 1 Pelabelan Data

Setelah melakukan pelabelan data, selanjutnya dilakukan pembagian dataset. Pada penelitian ini, proses pembagian data *training* dan data *testing* dilakukan dengan tiga skala yang berbeda. Skala data *training* dan data *testing* pertama yaitu 70:30, pada skala pertama ini jumlah data *training* 1050 gambar dan data *testing* 448 gambar. Skala pembagian data *training* dan data *testing* yang kedua yaitu 80:20, pada skala pembagian kedua ini jumlah data *training* 1197 gambar dan data *testing* 301 gambar. Skala pembagian data *training* dan data *testing* yang ketiga yaitu 90:10, dari skala pembagian ketiga ini jumlah data *training* 1348 gambar dan data *testing* 150 gambar.

Selanjutnya dilakukan proses augmentasi data pada data *training*, proses augmentasi data merupakan proses penambahan data yang bertujuan untuk meningkatkan akurasi model [10]. Proses ini dilakukan di *website* yang sama saat melakukan labeling yaitu Roboflow. proses augmentasi yang dilakukan yaitu *resize* gambar menjadi ukuran 320x320 piksel, *horizontal flip*, *vertical flip*, dan rotasi dengan sudut 15°.

D Modelling

Pada tahapan ini dilakukan proses *training* atau pelatihan deteksi penyakit kulit kucing dengan YOLOv8. Pada proses ini dilakukan *Hyperparameter Tuning*, parameter yang akan di setting untuk *hyperparameter* yaitu *learning rate*, *optimizer*, *epoch*, dan *batch size*. Pada Tabel I akan ditampilkan nilai-nilai parameter untuk *hyperparameter tuning*.

TABEL I
PARAMETER TRAINING

ss	Nilai
Epoch	50 dan 100
Batch size	16, 32, 64
Learning rate	0.1 – 0.00001
optimizer	SGD, Adam, AdamW, NAdam, RAdam, RMS Prop

Pada proses *training* ini, menggunakan *pretrained weight* “Yolov8n.pt”. Proses *Training* akan dilakukan di platform Google Collaboratory. Sebelum melakukan training

akun google collaboratory disambungkan dulu dengan akun gdrive, hal ini dilakukan untuk mengantisipasi hilangnya data hasil training jika terjadi error atau runtime yang terputus. Langkah pertama untuk training dengan algoritma YOLOv8 yaitu mengcloning repository github YOLOv8, hal ini dilakukan agar bisa menambahkan parameter baru untuk mengatasi imbalance class.

```
!git clone
https://github.com/ultralytics/ultralytics
!cd ultralytics
!pip install -e .
```

Gbr. 2 Code Melakukan Cloning YOLOv8

Pada penelitian ini dilakukan Teknik *class weight* untuk mengatasi *imbalance class*. *Class weight* adalah metode yang bisa diterapkan dalam *machine learning* untuk menangani ketidakseimbangan kelas dengan memberikan tingkat bobot yang lebih tinggi kepada kelas yang minoritas [11]. *Class weight* pada penelitian ini yaitu kelas jamur sebesar 5, kelas lain sebesar 9, kelas scabies sebesar 3, dan kelas sehat sebesar 3.

Setelah selesai mengatasi *imbalance class*, dilanjutkan proses *training*. Pada Gbr. 3 menjelaskan konfigurasi parameter untuk melakukan *training* dengan *pretrained model* *yolov8n.pt*, 50 *epoch*, *image size* 320 piksel, *batch size* 16, dan *optimizer* auto. *Optimizer* auto ini berfungsi untuk memilih *optimizer* secara otomatis yang disesuaikan dengan karakter data. Selain itu dengan menggunakan *optimizer* auto, *learning rate* juga akan otomatis disesuaikan agar meningkatkan performa model.

```
!yolo train data=data.yaml model=yolov8n.pt
epochs=50 imgsz=320 batch=16 optimizer='auto'
```

Gbr. 3 Code Training dengan YOLOv8

Proses *hyperparameter* ini dilakukan secara manual satu per satu. Setelah semua parameter sudah dicoba, uji coba selanjutnya yaitu berdasarkan pembagian data *training* dan data *testing*.

E Evaluation

Evaluasi performa model deteksi penyakit kulit kucing dengan YOLOv8 akan dilakukan dengan menghitung *precision* yang merupakan , *recall*, *mAP* dan *F1 score*.

$$Precision = \frac{TP}{TP+FP} \quad (1)$$

$$Recall = \frac{TP}{TP+FN} \quad (2)$$

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \quad (3)$$

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

Keterangan :

TP = True Positive

FP = False Positive

FN = False Negative

AP = Average Precision

Precision merupakan korelasi prediksi *true positive* dengan semua predisksi positive , *recall* merupakan proporsi prediksi positif secara akurat untuk semua data positif yang benar [12] . *F1-score* merupakan rata-rata gabungan dari *precision* dan *recall* [13] . *mAP* merupakan indikator utama dalam mengevaluasi kinerja model, Map ini adalah rata-rata dari Average precision (AP) [14].

F Development

Deployment merupakan tahapan akhir dari penelitian ini. *Deployment* merupakan tahapan dimana peneliti melakukan implementasi sistem agar menjadi aplikasi yang bisa digunakan oleh pengguna.proses *deployment* dilakukan dengan menggunakan Bahasa pemrograman Kotlin, sedangkan *testing* aplikasi menggunakan *white box testing*, *black box testing* dan uji analisis.

III. HASIL DAN PEMBAHASAN

A Hasil Training dan evaluasi

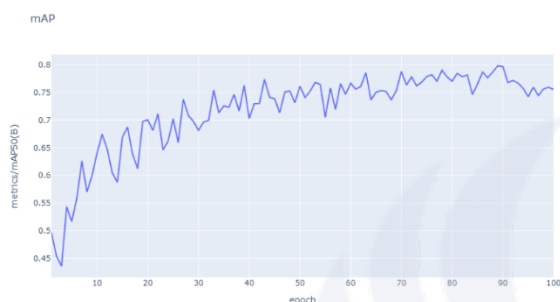
Proses *training* model yang dirancang dibagi menjadi beberapa bagian berdasarkan hyperparameter dan skala pembagia data yang suddah ditentukan sebelumnya. Adapun hasil training dari masing-masing data dapat dilihat pada Tabel II.

TABEL III
HASIL TRAINING

Skala dataset	Epoch	Batch Size	mAP	Precision	Recall	F1 Score
70:30	50	16	0.584	0.642	0.589	0.606
70:30	50	32	0.578	0.629	0.602	0.615
70:30	50	64	0.563	0.646	0.52	0.576
70:30	100	16	0.587	0.608	0.59	0.599
70:30	100	32	0.611	0.64	0.59	0.614
70:30	100	64	0.571	0.629	0.558	0.591
80:20	50	16	0.696	0.706	0.645	0.674
80:20	50	32	0.681	0.678	0.641	0.659
80:20	50	64	0.691	0.723	0.658	0.689
80:20	100	16	0.701	0.726	0.626	0.672
80:20	100	32	0.702	0.02	0.648	0.674
80:20	100	64	0.682	0.757	0.611	0.676
90:10	50	16	0.779	0.671	0.755	0.711
90:10	50	32	0.776	0.74	0.704	0.722
90:10	50	64	0.778	0.778	0.686	0.729
90:10	100	16	0.788	0.727	0.769	0.747

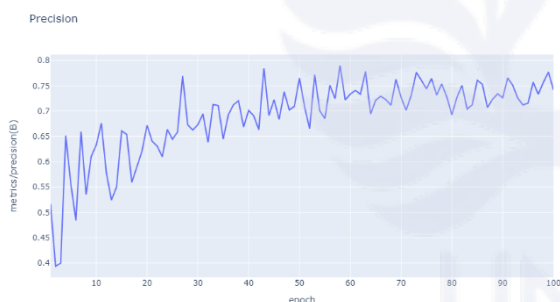
90:10	100	32	0.758	0.751	0.684	0.716
90:10	100	64	0.766	0.742	0.717	0.729

Pada Tabel II Didapatkan hasil paling optimal dari keseluruhan *experiment* yaitu dengan menggunakan *epoch* 100, *batch size* 16, *optimizer* AdamW, *learning rate* 0.00125, dan dengan skala pemabgain dataset 90% data *training* dan 10% data *testing*. Hasil *experiment* terbaik didapatkan *score* mAP yang bernilai 0.788, *precision* yang bernilai 0.727, *recall* yang bernilai 0.769, dan F1-Score yang berniali 0.747. Berikut grafik-grafik *training* model.



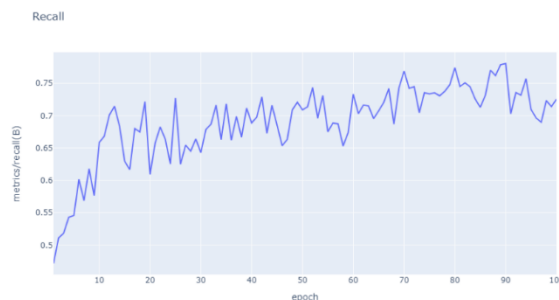
Gbr. 4 mAP

Selain grafik Map, berikut merupakan grafik *precision* selama masa pelatihan model deteksi penyakit kulit kucing. Pada grafik tersebut bisa dilihat bahwa selama masa pelatihan atau *training* menggambarkan peningkatan nilai.



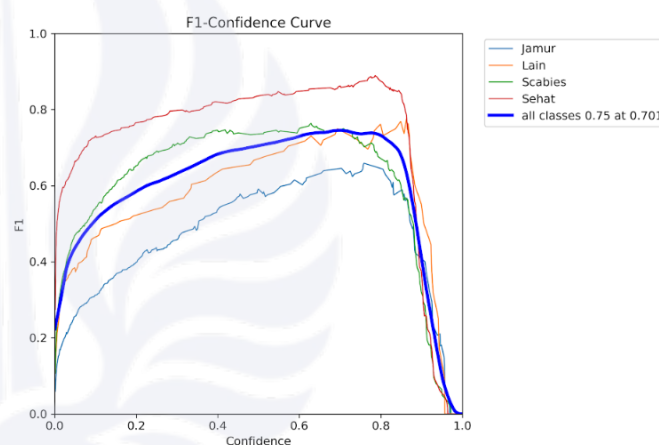
Gbr. 5 Precision

Berikut merupakan grafik *recall* selama masa pelatihan model deteksi penyakit kulit kucing. Pada grafik tersebut bisa dilihat bahwa selama masa pelatihan atau *training* menggambarkan peningkatan nilai.



Gbr. 6 Recall

Berikut merupakan grafik F1-Score selama masa pelatihan model deteksi penyakit kulit kucing. Pada grafik tersebut tersaji bahwa selama masa pelatihan atau *training* menggambarkan peningkatan nilai.



Gbr. 7 F1-Score

Tahapan selanjutnya yaitu *interpretation/ evaluation*, pada tahapan ini best model yang sudah tersimpan akan digunakan untuk evaluasi data testing, code untuk melakukan evaluasi bisa dilihat pada Gbr. 8.

```
!yolo val
model=runs/detect/train/weights/best.pt
data=data.yaml
```

Gbr. 8 Code Evaluasi YOLOv8

Pada Tabel III merupakan hasil evaluasi model didapatkan hasil testing yang baik, dimana hasil training dan testing cenderung menghasilkan *score* mAP yang sama yaitu 0.788. pada penelitian yang telah dilakukan menghasilkan model deteksi yang cukup bagus sehingga model deteksi bisa diimplementasikan menjadi aplikasi deteksi penyakit kulit kucing. Kesalahan deteksi disebabkan oleh gambar yang kurang jelas dan jauh karena sumber data yang bervariasi.

Selain itu, kelashan deteksi juga dipengaruhi dari kesulitan pengumpulan data sehingga data yang digunakan kurang bervariasi.

TABEL IIIII
HASIL EVALUASI

Process	mAP	Precision	Recall	F1 Score
Validasi/Testing	0.788	0.727	0.769	0.747
Train	0.788	0.727	0.769	0.747

Setelah mengevaluasi model deteksi penyakit kulit kucing, selanjutnya yait mengexport model ke format tflite. Export model ini dilakukan untuk proses deployment, pada Gbr. 9 merupakan kode untuk melakukan export model.

```
!yolo export
model=runs/detect/train/weights/best.pt
format=tflite int8=True
```

Gbr. 9 Code Export Model YOLOv8

B Implementasi sistem

Pada tahapan ini dilakukan proses implementasi model deteksi penyakit kulit kucing yang sudah dibuat sebelumnya. Adapaun user dari aplikasi ini adalah *cat owner* / pemelihara kucing. Adapun fitur yang dibutuhkan untuk aplikasi deteksi penyakit kulit kucing terdiri dari fitur kamera untuk deteksi penyakit kulit kucing secara *real time*, fitur import gambar untuk deteksi penyakit kulit kucing dari gambar, dan fitur penyakit kulit yang akan berisi informasi mengenai penyakit kulit kucing, ciri-ciri penyakit, tips penanganannya, dan dilengkapi degan cara merawat kucing sehat dan penyakit lainnya yang berisi informasi singkat mengenai berbagai jenis penyakit kulit kucing selain jamur dan scabies. Berikut merupakan hasil implementasi sistem:

a. Halaman Awal

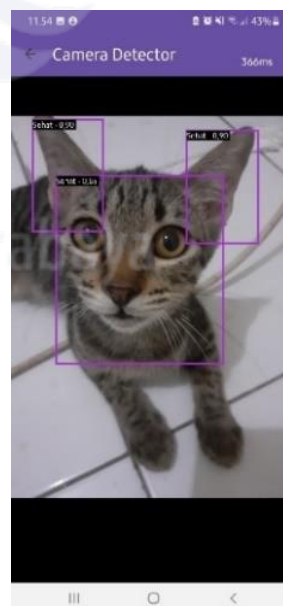
Halaman awal ini merupakan tampilan yang akan muncul pertama kali saat membuka aplikasi. Pada tampilan ini terdapat beberapa *button* menu yaitu, camera detector, import image, dan tentang penyakit.



Gbr. 10 Tampilan awal

b. Camera Detector

Pada halaman camera detector ini user bisa melakukan *realtime* deteksi penyakit kulit kucing dengan mengarahkan kamera ke kucing atau kearah posisi yang diduga ada penyakitnya. Namun, pada fitur *camera detector* ini bisa mendeteksi objek apapun selain kucing jika memiliki karakteristik yang sama seperti penyakit yang bisa dideteksi di aplikasi ini. Karakteristik yang dimaksud yaitu adanya kesamaan tekstur, bentuk, atau warna. Selain itu pada fitur ini juga tidak bisa melakukan *zoom in* maupun *zoom out*.



Gbr. 11 Camera Detector

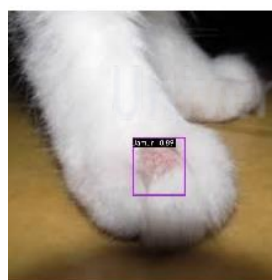
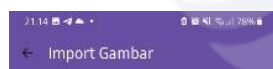
c. Import Image

Pada halaman ini user digunakan untuk mendeteksi penyakit kulit kucing dengan import gambar, untuk melakukan pendeteksian user harus memasukkan gambar untuk dideteksi dengan klik gambar “click here”.



Gbr. 12 Import Gambar

Tampilan setelah memilih gambar akan menjadi seperti Gbr. 13 Untuk memasukkan gambar lagi, user bisa klik gambarnya.



Gbr. 13 Tampilan Hasil Import Gambar

d. Tentang Penyakit

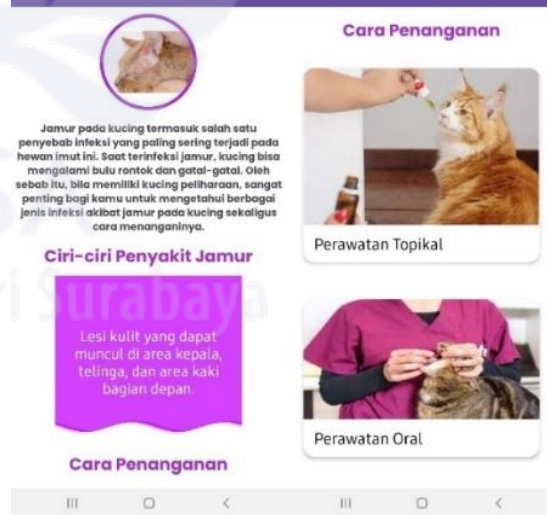
Pada halaman tentang penyakit ini berisi sub menu informasi mengenai penyakit kulit kucing diantaranya jamur, scabies, sehat, dan penyakit lainnya.



Gbr. 14 Tampilan Tentang Penyakit

1. Jamur

Pada halam ini user bisa melihat informasi seputar penyakit jamur pada kucing seperti apa itu jamur, ciri-ciri jamur pada kucing, dan cara penanganannya.



(a)

(b)

Gbr. 15 (a) dan (b) tampilan jamur

Pada gambar di bawah ini merupakan tampilan-tampilan cara perawatan jamur pada kucing. Pada aplikasi ini cara perawatan jamur dibagi menjadi tiga acara yaitu perawatan topikal, perawatan oral, dan perawatan lingkungan. Adapun

gambar perawatan topikal bisa dilihat pada Gbr.16 . Pada tampilan ini user bisa melihat cara perawatan kucing yang sedang mengalami penyakit jamur dengan cara pengobatan topikal.



Gbr. 16 Tampilan Perawatan Topikal

Berikut merupakan tampilan perawatan oral jamur. Pada tampilan ini user bisa melihat cara perawatan kucing yang sedang mengalami penyakit jamur dengan penobatan oral.



Gbr. 17 Tampilan Perawatan Oral

Berikut merupakan tampilan perawatan lingkungan jamur. Pada tampilan ini user bisa melihat cara perawatan lingkungan kucing yang sedang mengalami penyakit jamur

agar penyakit itu tidak menyebarkan ke kucing yang lain ataupun kepemilik kucing.



Gbr. 18 Tampilan Perawatan Lingkungan

2. Scabies

Pada halaman ini user bisa melihat informasi seputar penyakit scabies pada kucing seperti apa itu penyakit scabies pada kucing, ciri-ciri peyakit scabies pada kucing, dan cara penanganannya.



Gbr. 19 Tampilan Scabies

Pada Gbr. 20 merupakan tampilan Ketika user mengklik button “cara pengobatan” di halaman scabies, pada tampilan

ini berisi informasi cara penanganan penyakit scabies pada kucing dan juga rekomendasi obat untuk membantu penyembuhan penyakit scabies pada kucing.



Gbr. 20 Tampilan Pengobatan Scabies

3. Penyakit Lainnya

Pada halaman ini user bisa melihat informasi seputar penyakit lain diluar jamur dan scabies. Pada halaman ini informasi yang didapat user tidak sebanyak penyakit jamur, scabies maupun sehat, karena pada halaman ini user hanya bisa melihat apa saja penyakit kulit pada kucing.



Gbr. 21 Tampilan Penyakit Lain

4. Sehat

Pada halaman ini user bisa melihat informasi seputar cara merawat agar kucing agar tetap sehat dan tidak mudah terkena penyakit kulit kucing.



Gbr. 22 Tampilan Sehat

C Pengujian Sistem

Setelah pembuatan aplikasi selesai dilakukan pengujian aplikasi apakah aplikasi bisa berjalan dengan baik dan bisa mendeteksi dengan benar. Proses pengujian aplikasi menggunakan metode *white box testing*, *black box testing* dan uji analisis dengan menggunakan tiga smartphone yang berbeda.

a. White box testing

Pada pengujian *white box testing* dilakukan dengan menerapkan Teknik *basis path*. *Basis Path Testing* merupakan salah satu metode pengujian *white box testing*, yang dalam prosesnya memerlukan pembuatan flow graph dari program script serta penentuan nilai *cyclomatic complexity*. Tes ini bertujuan untuk menganalisis kebenaran struktur program dan kinerja program [15]. Pengujian ini dilakukan pada setiap fitur yang ada pada aplikasi, berikut merupakan hasil rekapitulasi *white box testing*:

TABEL IV
WHITE BOX TESTING

Fitur	Jumlah Independent Path	Hasil	
		Benar	Salah
Camera Detector	2	2	0
Import Gambar	2	2	0
Tentang Penyakit	6	6	0

Hasil pengujian *white box* pada tabel di atas dengan menggunakan metode *basis path* menunjukkan bahwa seluruh *test case* berhasil dieksekusi minimal satu kali. Berdasarkan hasil tersebut dapat disimpulkan bahwa semua fitur pada aplikasi deteksi penyakit kulit kucing ini lolos uji *white box*.

b. *Black box testing*

Pada pengujian ini dilakukan dengan menjalankan scenario pengujian, proses pengujian dilakukan dengan menggunakan lima *smartphone*. Berikut pada Tabel V merupakan daftar *smartphone* yang digunakan.

TABEL V
DAFTAR SMARTPHONE

Smartphone	Resolusi Kamera
Samsung A10	13 mp
Samsung A32	64 mp
Vivo Y02	8 mp
Samsung A04	50 mp
OPPO A53	13 mp, 2 mp, 2mp

Pada Tabel V Merupakan daftar Smartphone yang akan digunakan untuk menguji aplikasi deteksi penyakit kulit kucing. Pada proses *black box testing*, nama *smartphone* akan dikodekan menjadi I untuk Samsung A10, II untuk Samsung A23, III untuk vivo YO2, IV Samsung A04, dan V untuk OPPO A53 . Berikut pada Tabel VI hasil dari *black box testing*. Berikut merupakan scenario pengujian *black box* :

TABEL VI
SKENARIO BLACK BOX TESTING

NO	Skenario Uji	Hasil diharapkan
1	Membuka Aplikasi	Menampilkan halaman awal.
2	Mengklik menu camera detector	Menampilkan camera detector dan hasil deteksi secara realtime
3	Mengklik menu import gambar	Menampilkan halaman import gambar
4	Mengklik gambar “click here”	Menampilkan galeri untuk upload gambar dan menampilkan hasil deteksi
5	Mengklik menu tentang penyakit	Menampilkan halaman tentang penyakit
6	Mengklik submenu jamur	Menampilkan halaman jamur
7	Mengklik cara penanganan oral	Menampilkan halaman penanganan oral
8	Mengklik cara penanganan topikal	Menampilkan halaman penanganan topical
9	Mengklik cara penanganan lingkungan	Menampilkan halaman cara penanganan

		lingungan
10	Mengklik submenu scabies	Menampilkan halaman scabies
11	Mengklik button “cara penanganan”	Menampilkan halaman cara penanganan
12	Mengklik submenu penyakit lainnya	Menampilkan halaman penyakit lainnya
13	Mengklik submenu sehat	Menampilkan halaman sehat
14	Mengklik button back	Kembali kehalaman sebelumnya

Berikut hasil rekapitulasi *black box testing* pada lima jenis *smartphone*:

TABEL VII
HASIL BLACK BOX TESTING

Skenario uji ke -	Smartphone				
	I	II	III	IV	V
1	✓	✓	✓	✓	✓
2	✓	✓	✓	✓	✓
3	✓	✓	✓	✓	✓
4	✓	✓	✓	✓	✓
5	✓	✓	✓	✓	✓
6	✓	✓	✓	✓	✓
7	✓	✓	✓	✓	✓
8	✓	✓	✓	✓	✓
9	✓	✓	✓	✓	✓
10	✓	✓	✓	✓	✓
11	✓	✓	✓	✓	✓
12	✓	✓	✓	✓	✓
13	✓	✓	✓	✓	✓
14	✓	✓	✓	✓	✓

Dari hasil pengujian dengan *black box testing* didapatkan bahwa semua fitur bisa digunakan pada kelima jenis *smartphone*

c. Uji Analisis

Uji analisis dilakukan untuk menguji lebih dalam fitur pendeteksian aplikasi deteksi penyakit kulit kucing dengan berbagai kondisi maupun jarak untuk mendeteksi penyakit kulit kucing dan juga menguji konsistensi pendeteksian pada fitur import image. Pada uji analisis ini menggunakan 100 data uji yang diujikan pada lima *smartphone* yang berbeda. Berikut merupakan hasil rekapitulasi uji analisis pada fitur camera detector:

TABEL VIII
UJI CAMERA DETECTOR

Smartphone	Akurasi pengujian			Rata-rata
	5 CM	10 CM	20 CM	
Samsung A10	0.75	0.86	0.89	0.83
Samsung A32	0.91	0.96	0.97	0.95
Vivo Y02	0.72	0.82	0.87	0.80
Samsung A04	0.83	0.90	0.94	0.89
OPPO A53	0.90	0.94	0.96	0.93
Rata-rata	0.82	0.89	0.92	

Dari hasil uji camera detector dengan jarak 5 cm, 10 cm dan 20 cm didapat pengujian terbaik yaitu pada jarak 20 cm dengan menggunakan *smartphone* Samsung A32 dengan nilai akurasi 0.97. Dari pengujian ini didapatkan rata-rata akurasi sebesar 0.92. dari hasil tersebut aplikasideteksi penyakit kulit kucing paling optimal dilakukan pada jarak 20 cm dari objek yang dideteksi.

Selanjutnya yaitu pengujian pada fitur import gambar, pengujian ini juga dilakukan pada 100 data gambar. Adapun hasil pengujian bisa dilihat pada Tabel IX

TABEL VIIIIX
UJI IMPORT GAMBAR

Smartphone	Jumlah Gambar	Hasil	
		Benar	Salah
Samsung A10	100	92	8
Samsung A32	100	92	8
Vivo Y02	100	92	8
Samsung A04	100	92	8
OPPO A53	100	92	8

Dilihat pada Tabel IX hasil pengujian pada fitur uji import gambar di *smartphone* yang berbeda mencapai akurasi yaitu 0.92, dari hasil ini dapat dikatakan pendeteksian penyakit kulit kucing dengan fitur import gambar konsisten dilima *smartphone* yang berbeda.

IV. KESIMPULAN

Pada penelitian ini, peneliti menggunakan algoritma YOLOv8 dengan arsitektur YOLOv8n untuk mendeteksi penyakit kulit kucing yaitu jamur, scabies, penyakit kulit lain, dan sehat. Dataset yang digunakan dalam penelitian ini berjumlah 1498, data ini dibagi menjadi 3 skala data *training* dan data *testing* yang berbeda yaitu 70:30, 80:20, dan 90:10. Pelatihan dilakukan dengan melakukan uji coba

hyperparameter, hasil uji coba mencapai score Map 0.788, F1- score 0.75, *precision* 0.727, *recall* 0.769 dengan parameter terbaik yaitu *epoch* 100, *batch size* 16, *optimizer* AdamW, dan *learning rate* 0.00125.

Dari hasil pengembangan aplikasi deteksi penyakit kulit kucing berbasis android didapatkan kesimpulan bahwa model deteksi yang sudah dibuat dapat mendeteksi penyakit kulit kucing. Hasil Pengujian *white box* berhasil berjalan sesuai *test case* yang ada. Hasil pengujian *black box* yaitu semua fitur aplikasi dapat berjalan dengan baik dilima *smartphone* yang berbeda. Hasil uji analisis dalam pengujian fitur *camera detector* didapatkan bahwa aplikasi deteksi penyakit kulit kucing paling optimal dilakukan pada jarak 20 cm dengan skor akurasi pengujian sebesar 0.92. Hasil uji ini juga mampu mendeteksi penyakit kulit kucing dengan resolusi kamera yang berbeda beda. Hasil uji analisis dalam pengujian fitur import gambar didapatkan bahwa fitur ini mampu mendeteksi secara konsisten dilima *smartphone* yang berbeda, didapatkan akurasi pengujian sebesar 0.92.

V. SARAN

Berikut beberapa saran untuk pengembangan penelitian ini :

- Memperbanyak variasi gambar, variasi yang dimaksud seperti resolusi gambar, variasi sudut objek, variasi waktu pengambilan gambar (siang atau malam). Hal ini bertujuan agar model deteksi bisa lebih optimal dalam berbagai kondisi.
- Memperbanyak kelas penyakit yang akan dideteksi. Dengan memperbanyak kelas penyakit diharapkan bisa model deteksi jauh lebih akurat.
- Menggunakan algoritma deteksi objek yang lain. Dengan menggunakan algoritma yang berbeda diharapkan akan ada perbedaan yang signifikan.
- Menambahkan database pada aplikasi yang akan dibuat.

REFERENSI

- Nurheti Yulianti, *Hidup Sehat Bersama KUCING Kesayangan*. Gramedia Pustaka Utama, 2013.
- "Peluang Bisnis Hewan Peliharaan di Indonesia - Kompas.id." Accessed: Nov. 08, 2023. [Online]. Available: https://www.kompas.id/baca/riset/2023/08/03/peluang-bisnis-hewan-peliharaan-di-indonesia?status=sukses_login&status_login=login
- A. Menggunakan Metode Teorema Bayes di Balai Pelayanan dan Pengujian Veteriner Kabupaten Serang Berbasis Web Sri setiyowati, A. heriwibowo, N. nailul wardah, A. gilar pratama, F. Teknologi dan Informatika universitas Mathla, and ul Anwar Banten, "Sistem Pakar Diagnosa Penyakit Kulit pada Kucing Ras," 2019.
- I. Gunawan and R. Febryansyah, "PENERAPAN METODE NAIVE BAYES PADA SISTEM PAKAR DIAGNOSA PENYAKIT KULIT KUCING BERBASIS WEB."
- C. Widiyawati and M. Imron, "Sistem Pakar Diagnosa Penyakit Pada Kucing Menggunakan Metode Naive Bayes Classifier Expert System Of Cat Disease Diagnosis Using Naive Bayes Classifier Method," 2018.
- I. Yoga Pangestu and S. R. Ramadhani, "Perancangan Sistem Deteksi Penyakit Kulit Pada Kucing Menggunakan Deep Learning

- Berbasis Android,” vol. 12, no. 3, pp. 173–182, 2023, doi: 10.34148/teknika.v12i3.673.
- [7] K. Maulana Azhar, I. Santoso, D. Yosua, and A. A. Soetrisno, “Implementasi Deep Learning Menggunakan Metode Convolutional Neural Network Dan Algoritma Yolo Dalam Sistem Pendeteksi Uang Kertas Rupiah Bagi Penyandang Low Vision,” 2021. [Online]. Available: <https://ejournal3.undip.ac.id/index.php/transient>
- [8] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE Computer Society, Dec. 2016, pp. 779–788. doi: 10.1109/CVPR.2016.91.
- [9] L. Firgia and S. Thomas, “Deteksi Jenis Penyakit Dan Hama Pada Tanaman Jagung Menggunakan Arsitektur Spatial Pyramid Pooling Pada YOLOv5s,” *Jurnal Riset Sistem Informasi Dan Teknik Informatika (JURASIK)*, vol. 8, no. 2, pp. 452–459, 2023, [Online]. Available: <https://tunasbangsa.ac.id/ejurnal/index.php/jurasik>.
- [10] R. H. Jatmiko and Y. Pristyanto, “Investigating The Effectiveness of Various Convolutional Neural Network Model Architectures for Skin Cancer Melanoma Classification,” *MATRIK : Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 23, no. 1, pp. 1–16, Oct. 2023, doi: 10.30812/matrik.v23i1.3185.
- [11] H. Akbar and W. K. Sanjaya, “Kajian Performa Metode Class Weight Random Forest pada Klasifikasi Imbalance Data Kelas Curah Hujan,” *Jurnal Sains, Nalar, dan Aplikasi Teknologi Informasi*, vol. 3, no. 1, Dec. 2023, doi: 10.20885/snati.v3i1.30.
- [12] E. Dio Bagus Sudewo, M. Kunta Biddini, A. Fadlil, E. D. Sudewo, M. Biddinika, and A. Fadlil, “DenseNet Architecture for Efficient and Accurate Recognition of Javanese Script Hanacaraka Character How to Cite: ‘DenseNet Architecture for Efficient and Accurate Recognition of Javanese Script Hanacaraka Character.’” vol. 23, no. 2, pp. 453–464, 2024, doi: 10.30812/matrik.v23i2.xxx.
- [13] N. W. S. Saraswati, I. W. D. Suryawan, N. K. T. Juniartini, I. D. M. K. Muku, P. Pirozmand, and W. Song, “Recognizing Pneumonia Infection in Chest X-Ray Using Deep Learning,” *MATRIK : Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 23, no. 1, pp. 17–28, Oct. 2023, doi: 10.30812/matrik.v23i1.3197.
- [14] X. Chen, J. Lv, Y. Fang, and S. Du, “Online Detection of Surface Defects Based on Improved YOLOV3,” *Sensors*, vol. 22, no. 3, Feb. 2022, doi: 10.3390/s22030817.
- [15] I. G. S. Rahayuda and N. P. L. Santiari, “Basis Path Testing Of Iterative Deepening Search And Held-Karp On Pathfinding Algorithm,” *Jurnal Ilmiah Kursor Menuju Solusi Teknologi Informasi*, 2017.