

Implementasi *Computer Vision* untuk Klasifikasi Gambar Kucing dan Anjing Menggunakan OpenCV-Python

Chaerul Mustofa^{1*}, Akbar Berwyn Kurniawan², Wahyu Afrinaldi³, Perani Rosyani⁴

¹⁻⁴Fakultas Ilmu Komputer, Teknik Informatika, Universitas Pamulang, Jl. Raya Puspipetek No. 46, Kel. Buaran, Kec. Serpong, Kota Tangerang Selatan. Banten 15310, Indonesia

Email: ^{1*}chaerulmustofa95@gmail.com, ²akbarberwyn@gmail.com,

³wahyuafrialdi014@gmail.com, ⁴dosen00837@unpam.ac.id

(* : coressponding author)

Abstrak–Studi kasus ini peneliti membahas implementasi computer vision untuk klasifikasi gambar kucing dan anjing menggunakan OpenCV dan Python. Tujuan dari di buatnya studi kasus ini adalah untuk dapat mengembangkan model klasifikasi yang dapat membedakan antara gambar kucing dan anjing dengan tingkat akurasi yang tinggi. Proses dimulai dengan pengumpulan dataset dari gambar kucing dan anjing yang kemudian di proses untuk meningkatkan kualitas dari gambar . Teknik augmentasi data akan diterapkan untuk memperluas variasi dataset dan meningkatkan performa model. Algoritma Convolutional Neural Network (CNN) digunakan sebagai dasar model klasifikasi. Model CNN dilatih menggunakan dataset yang telah dipreproses dan divalidasi dengan menggunakan cross-validation untuk menghindari overfitting. OpenCV digunakan untuk menangani operasi gambar dasar seperti resizing, konversi warna, dan augmentasi data, sementara framework deep learning seperti TensorFlow untuk digunakan untuk membangun dan melatih model CNN. Hasil eksperimen akan menunjukkan bahwa model CNN dapat diimplementasikan mampu mencapai tingkat akurasi yang memuaskan dalam mengklasifikasikan gambar kucing dan anjing.

Kata Kunci: *Computer Vision*, Klasifikasi Gambar, OpenCV, Python, *Convolutional Neural Network*, Kucing, Anjing, Augmentasi Data, Pemrosesan Gambar

Abstract*In this case study, researchers discuss the implementation of computer vision for classifying images of cats and dogs using OpenCV Python. The aim of this study is to developing a classification model that can differentiate between images of cats and dogs with a high level of accuracy. The process begins with collecting dataset of a cat and a dog images which are then processed to improve the quality of the images. Data augmentation technique will be applied to expand the variety of datasets and improve model performance. The Convolutional Neural Network (CNN) algorithm is used as the basis for the classification model. The CNN model was trained using a preprocessed dataset and validated using cross-validation to avoid overfitting. OpenCV is used to handle basic image operations such as resizing, color conversion, and data augmentation, while deep learning frameworks such as TensorFlow are used to build and train CNN models. The experimental results will show that the implemented CNN model able to achieve a satisfactory level of accuracy in classifying images of cats and dogs.*

Keywords: *Computer Vision, Image Classification, OpenCV, Python, Convolutional Neural Network, Cats, Dogs, Data Augmentation, Image Processing*

1. PENDAHULUAN

Computer Vision adalah sebuah sistem yang memiliki tujuan untuk membuat komputer dapat menangkap sebuah informasi visual dari dunia nyata dengan menerapkan OpenCV. Teknologi ini membuat mesin dapat memproses gambar serta video dengan cara yang serupa seperti manusia. Salah satu alat yang digunakan dalam pengolahan gambar dan video adalah OpenCV, yang memiliki berbagai fungsi untuk memanipulasi dan menganalisis data visual. OpenCV tidak hanya berguna untuk pengolahan di gambar saja, tetapi OpenCV selalu digunakan dalam penelitian dan industri untuk pembuatan berbagai aplikasi, contohnya seperti scan wajah, scan objek, pengenalan pola, dan lain lainnya.

2. METODE PENELITIAN

Penggunaan metode dalam penelitian ini meliputi langkah langkah seperti berikut:

1. Mencari dan mengumpulkan data melalui jurnal jurnal yang tersedia
2. Membuat program untuk Klasifikasi Gambar Kucing dan Anjing
3. Melakukan *running test* pada program

2.1 Landasan Teori

Pengolahan citra merupakan sebuah metode yang berguna untuk memproses dua dimensi seperti foto atau gambar dengan menggunakan media komputer, citra memiliki bentuk seperti angka numerik yang berisi nilai-nilai secara diskrit atau disebut dengan nilai intensitas cahaya, yang memungkinkan komputer tersebut dapat mengolah visual secara digital.

Library merupakan kumpulan-kumpulan kode yang menyediakan fungsi fungsi tertentu untuk digunakan dalam aplikasi lain. Di dalam pemrograman *Python*, *library* memiliki beberapa *module* dan *package* yang di gabungkan untuk menyediakan sekumpulan fungsionalitas pada pemrograman *Python* dengan adanya *library* pada pemrograman *python* memungkinkan untuk menganalisa kode yang telah digunakan dan diuji sebelumnya,

Augmentasi data adalah proses menghasilkan data baru secara artifisial dari data yang ada, terutama untuk melatih model *machine learning* di karenakan Model *Machine learning* membutuhkan set data yang besar dan beragam untuk pelatihan awal Augmentasi data secara artifisial meningkatkan set data dengan membuat perubahan kecil pada data asli.

Tensorflow Memiliki fungsi untuk digunakan proses pengembangan penerapan mesin pembelajaran serta algoritma *Tensorflow* ini merupakan sebuah Software yang memiliki peluang untuk para programmer untuk bisa membuat grafik aliran data dengan menggunakan tampilan grafis atau serangkaian node yang diproses secara khusus.

Google Colab merupakan sebuah *platform* yang di gunakan untuk menjalankan atau berbagi kode Python melalui web browser. *Google colab* ini dirancang untuk dapat bekerja pada bagian *machine learning* dengan menyediakan lingkungan komputasi yang fleksibel dan mudah diakses.

2.2 Perancangan Perangkat

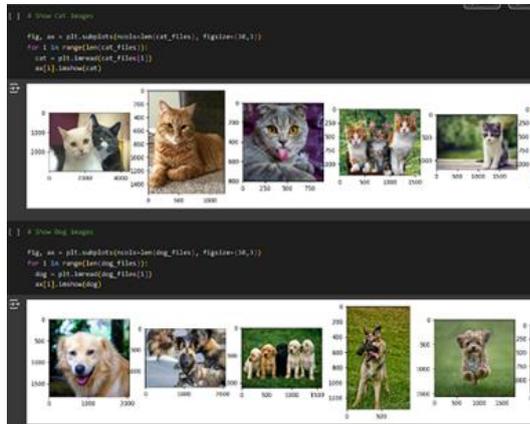
Dalam melakukan perancangan ini di gunakan nya bahasa pemrograman python OpenCV yang di proses melalui software pemrograman Google Colab gambar 1 ini merupakan tahapan awal dari langkah langkah dalam melakukan klasifikasi gambar pada Kucing dan Anjing.



Gambar 1. Langkah Awal Klasifikasi Gambar

Pada tahap awal ini merupakan Langkah-langkah bagaimana caranya melakukan klasifikasi, sebagai berikut:

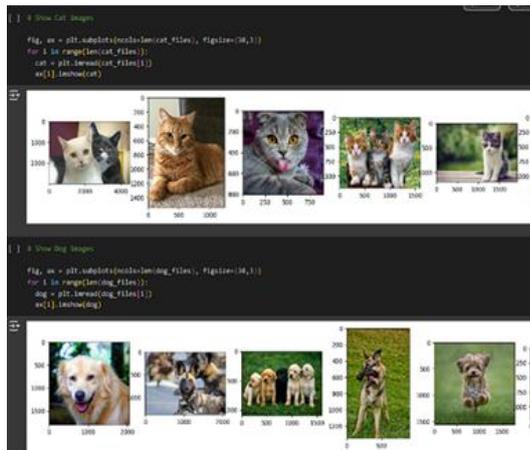
5. Menampilkan Gambar



Langkah terakhir adalah menampilkan gambar dengan memasukkan perintah untuk menampilkan kategori dari setiap gambar yang telah di masukkan.

3. ANALISA DAN PEMBAHASAN

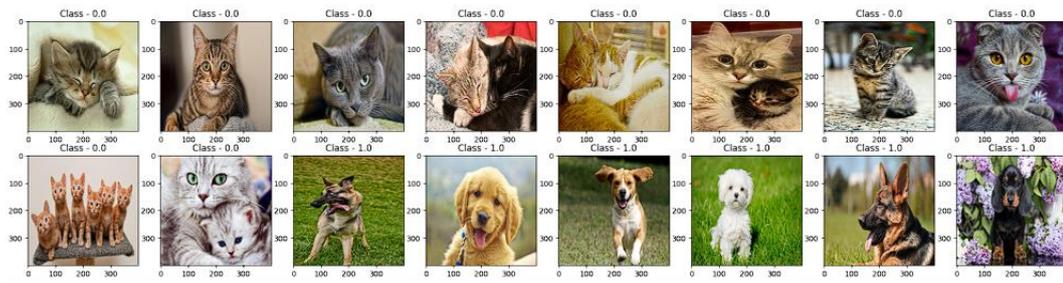
Hasil dari program klasifikasi yang di buat memiliki hasil untuk menampilkan kedua kategori gambar yang telah di *input*.



Gambar 2. Hasil Dari Program Klasifikasi Gambar Yang Telah di Input

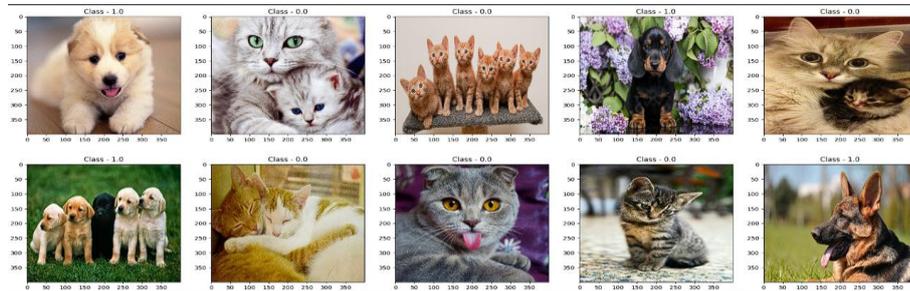
Tabel 1. Pengujian Perubahan Pada Gambar

Code Program	Penjelasan
<pre># Show each of image from both categories fig, ax = plt.subplots(nrows=2, ncols=8, figsize=(25,6)) n = 0 for i in range(2): for j in range(8): img = data[0][0][n].astype('uint8') ax[i][j].imshow(img) ax[i][j].set_title('Class - ' + str(data[0][1][n])) n += 1</pre>	<p>Kode ini menampilkan gambar dari 2 kategori yaitu, gambar kucing dan anjing secara bersamaan dengan range 2 untuk horizontal dan 8 untuk bertikal.</p>



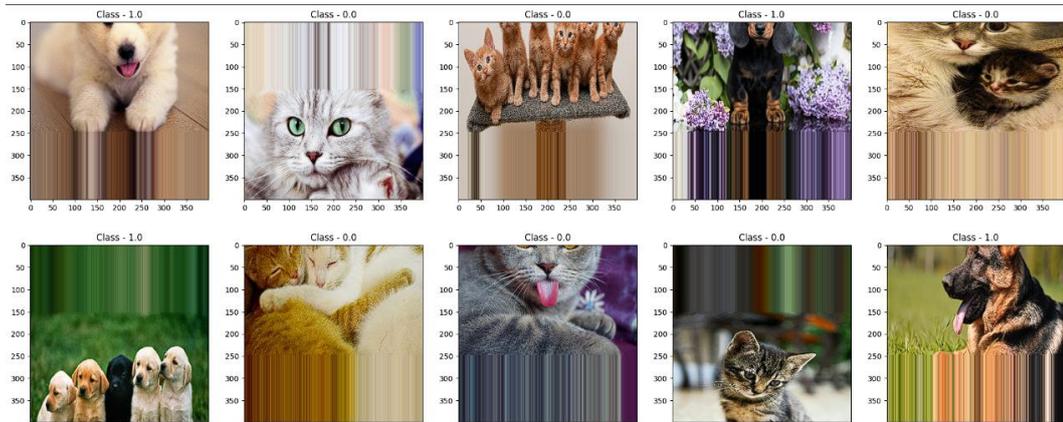
Tabel 2. Pengujian Perubahan Pada Gambar

Code Program	Penjelasan
<pre> # Show images of train-set fig, ax = plt.subplots(nrows=2, ncols=5, figsize=(25,10)) n = 0 for i in range(2): for j in range(5): img = train[0][0][n].astype('uint8') ax[i][j].imshow(img) ax[i][j].set_title('Class - ' + str(train[0][1][n])) n += 1 </pre>	<p>Kode ini bertujuan untuk menampilkan gambar dari dataset pelatihan (<i>train-set</i>) dalam bentuk grid menggunakan Matplotlib. Kode ini membuat grid 2x5 dan menampilkan 10 gambar dari dataset pelatihan, masing-masing dengan judul yang menunjukkan kelas gambar.</p>



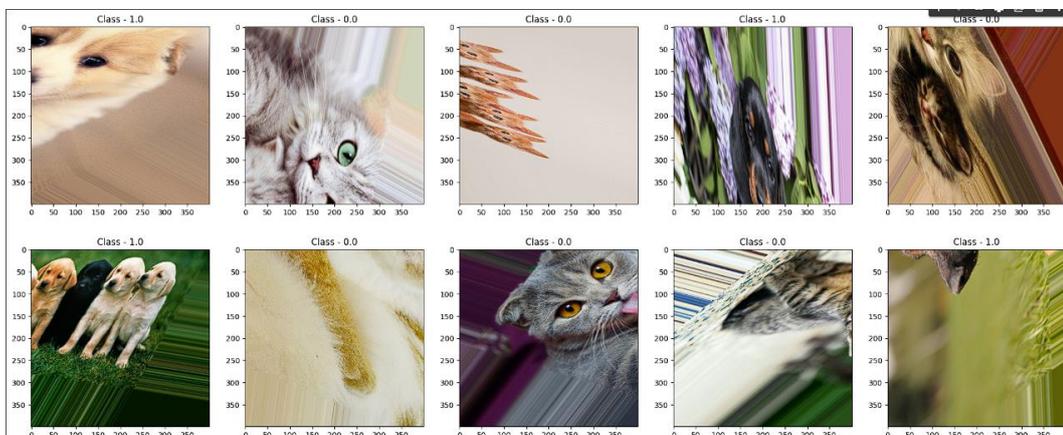
Tabel 3. Pengujian Perubahan Pada Gambar

Code Program	Penjelasan
<pre> # Data Augmentation - Width Shift imgen3 = tf.keras.preprocessing.image.ImageDataGenerator(width_shift_range=[-150, 150], validation_split=0.4) train3 = imgen3.flow_from_directory(path, target_size=(400,400), class_mode='binary', subset='training', shuffle=True, seed=2) val3 = imgen3.flow_from_directory(path, target_size=(400,400), class_mode='binary', subset='validation', shuffle=True, seed=2) # Show images of train-set fig, ax = plt.subplots(nrows=2, ncols=5, figsize=(25,10)) n = 0 for i in range(2): for j in range(5): img = train3[0][0][n].astype('uint8') ax[i][j].imshow(img) ax[i][j].set_title('Class - ' + str(train[0][1][n])) n += 1 </pre>	<p>Kode ini digunakan untuk membuat gambar yang dipilih memiliki menjadi tambah lebar ke dalam posisi horizontal.</p>



Tabel 4. Pengujian Perubahan Pada Gambar

Code Program	Penjelasan
<pre> # Data Augmentation - Combine all previous data augmentations imgen3 = tf.keras.preprocessing.image.ImageDataGenerator(width_shift_range=[-100, 100], height_shift_range=[-100, 100], rotation_range=120, zoom_range = [0.3, 1.5], shear_range=50, validation_split=0.4) train3 = imgen3.flow_from_directory(path, target_size=(400,400), class_mode='binary', subset='training', shuffle=True, seed=2) val3 = imgen3.flow_from_directory(path, target_size=(400,400), class_mode='binary', subset='validation', shuffle=True, seed=2) # Show images of train-set fig, ax = plt.subplots(nrows=2, ncols=5, figsize=(25,10)) n = 0 for i in range(2): for j in range(5): img = train3[0][0][n].astype('uint8') ax[i][j].imshow(img) ax[i][j].set_title('Class - ' + str(train3[0][1][n])) n += 1 </pre>	<p>Kode ini memiliki fungsi untuk menampilkan gambar dari gabungan seluruh teknik augmentasinya di mulai dari lebar ke kiri atau kanan atas bawah memperbesar gambar.</p>



4. KESIMPULAN

Studi kasus ini membuktikan bahwa teknik *computer vision* yang menggunakan OpenCV dan Python dapat secara efektif digunakan untuk mengklasifikasikan gambar kucing dan anjing. Proses ini melibatkan beberapa tahap penting, termasuk pengumpulan dan *preprocessing* data, augmentasi data, pengembangan dan pelatihan model *Convolutional Neural Network* (CNN), serta evaluasi performa model.

Dengan *preprocessing* dan augmentasi data yang tepat, model CNN yang dikembangkan mampu mencapai tingkat akurasi yang tinggi dalam membedakan gambar kucing dan anjing. Hal ini menunjukkan pentingnya langkah-langkah persiapan data dalam meningkatkan kemampuan model klasifikasi. Studi ini juga menggaris bawahi potensi besar *computer vision* dan *deep learning* dalam aplikasi praktis, seperti klasifikasi gambar binatang.

Secara keseluruhan, implementasi ini menyoroti kombinasi yang efektif antara teknik pemrosesan gambar dan algoritma *deep learning* untuk menyelesaikan tugas peneliti mengklasifikasi gambar dengan hasil yang memuaskan.

REFERENCES

- He, K., Zhang, X., Ren, S., & Sun, J. (2016). *Deep Residual Learning for Image Recognition*. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770-778.
- Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016). *SSD: Single Shot MultiBox Detector*. *Proceedings of the European Conference on Computer Vision (ECCV)*, 21-37.
- Van der Walt, S., Schönberger, J. L., Nunez-Iglesias, J., Boulogne, F., Warner, J. D., Yager, N., & Yu, T. (2014). *scikit-image: image processing in Python*. *PeerJ*, 2, e453.
- Google Colaboratory. (n.d.). Retrieved from <https://colab.research.google.com/>
- OpenCV. (n.d.). *Open Source Computer Vision Library*. Retrieved from <https://opencv.org/>
- Wibowo, A., & Anggoro, B. S. (2018). Deteksi dan Klasifikasi Objek Menggunakan OpenCV dan Python pada Robot *Line Follower*. *Jurnal Informatika: Jurnal Pengembangan IT*, 13(1), 12-20.
- Suhendra, I. (2020). Penerapan Augmentasi Data untuk Meningkatkan Akurasi Model *Deep Learning* dalam Klasifikasi Citra. *Seminar Nasional Teknologi Informasi dan Multimedia 2020*, Universitas Dian Nuswantoro, 45-52.
- Utami, T. R., & Kurniawan, R. (2017). Pemanfaatan Google Colaboratory untuk Pengembangan Model *Machine Learning*. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 1(4), 289-298.
- Handayani, D., & Prasetyo, E. (2020). Pemanfaatan *TensorFlow* untuk Pengenalan Pola Teks pada Dokumen Digital. *Jurnal Rekayasa Sistem dan Teknologi Informasi*, 4(2), 110-118.
- Nurhayati, N., & Supriyadi, T. (2018). Implementasi Teknik Data Augmentation pada Pengolahan Citra Digital untuk Peningkatan Akurasi Model *Deep Learning*. *Prosiding Seminar Nasional Informatika, 2018*, Universitas Pembangunan Nasional "Veteran" Yogyakarta, 54-61.
- Priyanto, S., & Hidayat, R. (2016). Pengenalan Karakter Menggunakan *Convolutional Neural Network* pada Platform Google Colab. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 2(3), 199-210.
- Perdana, W., Honi, R. A., Wibowo, B., & Rosyani, P. (2023). Studi Literature Review: Perbandingan Metode Klasifikasi Kecerdasan Buatan Pada Computer Vision. *JURIHUM: Jurnal Inovasi dan Humaniora*, 1(1), 192-195.
- ROSYANI, Perani; RETNAWATI, Retnawati. Ekstraksi Fitur Wajah Menggunakan Metode Viola Jones dengan Tools Cascade Detector. *JURIKOM (Jurnal Riset Komputer)*, 2023, 10.2: 633-639.
- Firmansyah, A., Itsnan, A. F., Apip, A., Mullia, R. T., & Rosyani, P. (2024). SISTEM ABSENSI MAHASISWA MENGGUNAKAN FACE RECOGNITION DENGAN ALGORITMA CNN. *AI dan SPK: Jurnal Artificial Intelligent dan Sistem Penunjang Keputusan*, 1(4).