

Implementasi Metode *Convolutional Neural Network* untuk Memilah Foto Pemandangan Pantai dan Pegunungan

Kalfin Syah Kilau Mayya¹, Irsyad Fadhil Makarim², Zenryo Yudi Arnava Darva Mahendra³,
Anggraini Puspita Sari^{4*}

^{1,2,3,4} Informatika, Universitas Pembangunan Nasional “Veteran” Jawa Timur

^{1,2,3} Jl. Rungkut Madya No.1, Gn. Anyar, Kec. Gn. Anyar, Surabaya, Jawa Timur, Indonesia, 60294

Email: ¹22081010251@student.upnjatim.ac.id, ²22081010131@student.upnjatim.ac.id,

³22081010303@student.upnjatim.ac.id, ^{4*}anggraini.puspita.if@upnjatim.ac.id

Submitted: 20/06/2024; Revised: 16/08/2024; Accepted: 21/08/2024

Abstrak— *Convolutional Neural Network* (CNN) merupakan salah satu metode yang dapat digunakan dalam klasifikasi gambar. Studi ini memaparkan implementasi CNN untuk memilah foto pemandangan pantai dan pegunungan. Masalah utama yang diangkat dalam penelitian ini adalah bagaimana mengembangkan model klasifikasi yang mampu membedakan antara gambar pemandangan pantai dan pegunungan, yang sering kali memiliki karakteristik visual yang mirip. Dataset terdiri dari 712 gambar pemandangan pantai dan 712 gambar pemandangan pegunungan yang kemudian dibagi menjadi data *train* dan *test*. Untuk memperkuat model, digunakan beberapa teknik augmentasi data seperti *resizing*, *shear*, *zoom*, dan *flip horizontal*. Model CNN dibangun dengan beberapa lapisan konvolusi, *pooling*, dan *fully connected* untuk ekstraksi fitur dan klasifikasi. Selama pelatihan, model menggunakan *binary crossentropy* sebagai fungsi *loss* dan *optimizer* Adam untuk mempercepat konvergensi model. Setelah proses pelatihan selama 20 *epoch* dengan *batch size* 32, model diuji menggunakan data *test* yang belum pernah dilihat oleh model sebelumnya. Hasil pengujian menunjukkan bahwa model CNN ini mampu mencapai akurasi 96.88% dalam data *train* dan 92.86% dalam data *test*. Hasil ini menunjukkan bahwa model CNN ini berhasil menghasilkan akurasi yang tinggi dalam membedakan gambar pemandangan pantai dan pegunungan berdasarkan dataset yang disediakan. Metode augmentasi data yang diterapkan seperti *resizing*, *shear*, *zoom*, dan *flip horizontal* membantu dalam meningkatkan keberhasilan model untuk membedakan gambar-gambar yang serupa namun berasal dari kategori yang berbeda.

Kata Kunci: *Convolutional Neural Network* (CNN); Klasifikasi gambar; Dataset; Pemrosesan Gambar; Pengenalan Pola; Deteksi Objek; Deep Learning; Pemandangan Alam; Pantai; Pegunungan; Algoritma Pembelajaran Mesin

Abstract— Convolutional Neural Network (CNN) is a method that can be used in image classification. This study describes the implementation of CNN to sort photos of beach and mountain views. The main problem raised in this research is how to develop a classification model that is able to differentiate effectively between images of coastal and mountain views, which often have similar visual characteristics. The dataset consists of 712 images of beach views and 712 images of mountain views which are then split into train and test data. To strengthen the model, several data augmentation techniques are used such as *resizing*, *shear*, *zoom*, and *horizontal flip*. The CNN model is built with several convolutional, *pooling*, and *fully connected* layers for feature extraction and classification. During training, the model uses *binary crossentropy* as a loss function and an *adam* optimizer to speed up convergence. After a training process of 20 *epochs* with a *batch size* of 32, the model was tested using test data that the model had never seen before. Test results show that this CNN model is able to achieve 96.88% accuracy in train data and 92.86% in test data. These results show that this CNN model succeeded in producing high accuracy in differentiating coastal and mountain scenery images based on the dataset provided. The applied data augmentation methods such as *resizing*, *shear*, *zoom*, and *horizontal flip* help in increasing the success of the model to differentiate images that are similar but come from different categories..

Keywords: Convolutional Neural Network (CNN); Image classification; Datasets; Image Processing; Pattern Recognition; Object Detection; Deep Learning; Panorama; Beach; Mountains; Machine Learning Algorithms

1. PENDAHULUAN

Dalam bidang fotografi, pemandangan pantai dan pegunungan merupakan objek yang menarik perhatian banyak fotografer [1]. Keindahan alam yang memukau sering menjadi inspirasi untuk mengabadikan momen-momen spesial melalui lensa kamera mereka [2]. Hubungan manusia dengan alam telah menginspirasi banyak individu untuk mengambil gambar pemandangan alam, baik saat matahari terbenam di pantai yang tenang atau pemandangan pegunungan yang menjulang tinggi dengan

latar belakang langit biru. Namun, para fotografer sering menghadapi kesulitan dalam kategorisasi hasil foto yang memiliki elemen campuran antara pemandangan pantai dan pegunungan, seperti foto yang menampilkan pantai dengan latar belakang gunung. Tantangan ini semakin besar ketika mereka harus memilah ribuan foto secara manual, yaitu dengan melihat dan mengidentifikasi setiap foto satu per satu untuk menentukan kategori yang tepat. Proses manual ini memakan banyak waktu dan tenaga, membuat pengelolaan koleksi foto menjadi tidak efisien dan sulit untuk diselesaikan dalam waktu singkat.

Keberagaman pemandangan alam, mulai dari garis pantai dengan ombak hingga pegunungan yang tertutup salju, menciptakan berbagai tantangan unik bagi para fotografer dalam hal komposisi, pencahayaan, dan warna. Meskipun keindahan tersebut memberikan kepuasan tersendiri bagi fotografer, namun proses pengorganisasian dan pengelompokan foto-foto tersebut seringkali menjadi tugas yang membutuhkan waktu yang lama. Terlebih lagi, bagi fotografer profesional yang bekerja dengan jumlah foto yang besar, proses manual untuk mengkategorikan foto-foto ini bisa sangat memakan waktu dan rentan terhadap kesalahan.

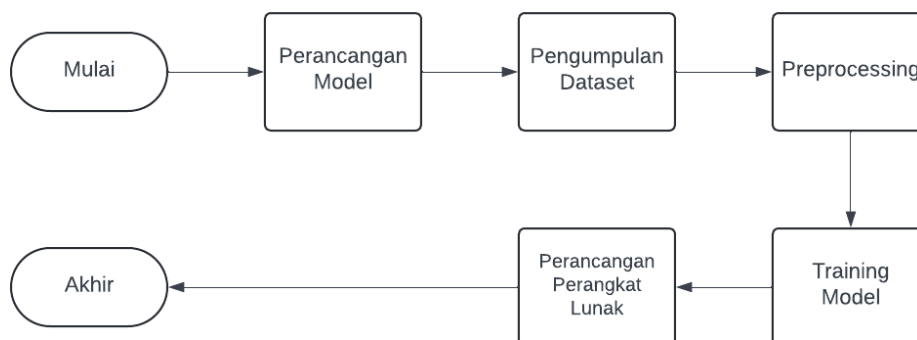
Guna mengatasi tantangan ini, teknologi telah menjadi solusi yang sangat potensial. Salah satu pendekatan yang efektif adalah dengan menggunakan metode *Convolutional Neural Network* (CNN). CNN adalah salah satu jenis jaringan saraf tiruan yang sangat efektif untuk pengenalan pola dalam gambar dan telah digunakan secara luas dalam berbagai aplikasi *computer vision*, seperti deteksi objek, pengenalan wajah, dan klasifikasi gambar [3]. Banyak penelitian telah menggunakan metode CNN untuk mengenali objek dalam gambar. Salah satu contohnya adalah dalam perancangan Self-Checkout System pada toko ritel, yang bertujuan untuk mempermudah dan mempercepat proses berbelanja [4].

Implementasi CNN dalam pengkategorian foto pemandangan pantai dan pegunungan melibatkan beberapa langkah penting. Hal pertama adalah pengumpulan dataset. Dataset merupakan kumpulan data yang disimpan dalam format yang terstruktur. Dataset dapat berupa kumpulan data numerik, teks, gambar, suara, atau jenis data lainnya yang digunakan untuk analisis atau pembelajaran mesin [5]. Dataset sering digunakan dalam berbagai bidang, termasuk ilmu data, pembelajaran mesin, dan statistik untuk menguji hipotesis, membuat prediksi, atau melakukan tugas analisis lainnya [6].

Dengan adanya program yang mengimplementasikan metode CNN ini, fotografer tidak lagi perlu menghabiskan banyak waktu untuk memilah-milah foto secara manual. Mereka dapat lebih fokus pada aspek kreatif dari fotografi, seperti eksplorasi komposisi dan pencahayaan, sementara tugas pengorganisasian foto dapat dilakukan secara otomatis dan lebih efisien oleh teknologi. Ini tidak hanya meningkatkan produktivitas, tetapi juga memastikan bahwa foto-foto disimpan dan diatur dengan cara yang lebih sistematis dan dapat diakses dengan mudah di masa mendatang. Dengan demikian, penggabungan teknologi CNN dalam dunia fotografi adalah langkah yang sangat positif. Selain membantu fotografer meningkatkan waktu efisiensi kerja, teknologi ini juga memungkinkan eksplorasi lebih dalam terhadap keindahan alam melalui fotografi. Dengan dukungan teknologi ini, diharapkan fotografer dapat terus menghasilkan karya-karya yang menginspirasi dan memukau, sementara teknologi mengambil peran penting untuk menjaga dan mengelola warisan visual alam secara lebih baik.

2. METODOLOGI PENELITIAN

A. Alur Penelitian

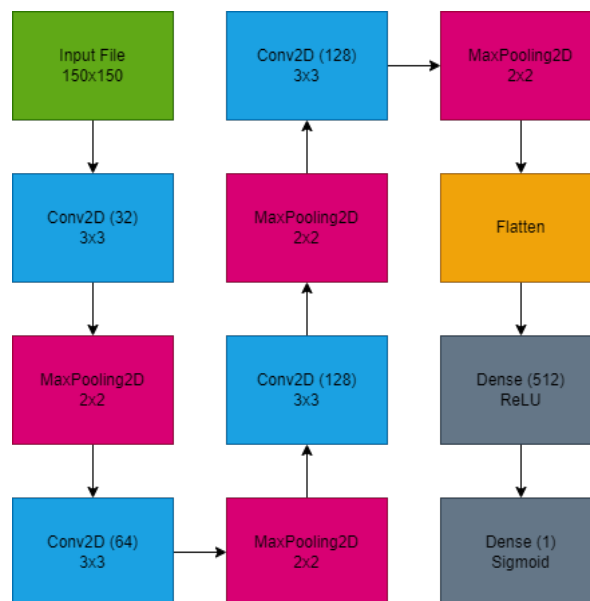


Gambar 1. Diagram Alur Penelitian

Alur penelitian ini dimulai dengan perancangan model, di mana struktur dan arsitektur model ditentukan. Selanjutnya, dilakukan pengumpulan dataset yang akan digunakan dalam proses pelatihan model. Setelah dataset terkumpul, data tersebut melalui tahap *preprocessing* untuk memastikan konsistensi, seperti penyesuaian ukuran gambar. Model kemudian dilatih menggunakan dataset yang telah dipreproses untuk mengenali pola dan melakukan klasifikasi. Setelah model mencapai kinerja yang memuaskan, tahap terakhir adalah perancangan perangkat lunak yang memungkinkan pengguna untuk mengklasifikasi foto pemandangan pantai dan pegunungan dengan model yang sudah dilatih.

B. Perancangan Model

Penelitian ini menggunakan CNN untuk menyelesaikan permasalahan klasifikasi foto pantai dan pegunungan. Model CNN yang dibangun terdiri dari total 9 lapisan: 4 lapisan konvolusi, 4 lapisan pooling, dan 1 lapisan flatten, diikuti oleh 2 lapisan *fully connected*. Model dilatih selama 20 *epoch*, dengan setiap *epoch* terdiri dari beberapa langkah pelatihan yang sesuai dengan jumlah *batch*. Proses augmentasi data pelatihan menggunakan beberapa teknik untuk meningkatkan generalisasi model dengan variasi tambahan pada data input. Hasil akhir dari model adalah klasifikasi biner yang memprediksi apakah gambar yang diberikan adalah gambar pemandangan pantai atau pegunungan.



Gambar 2. Diagram Model CNN

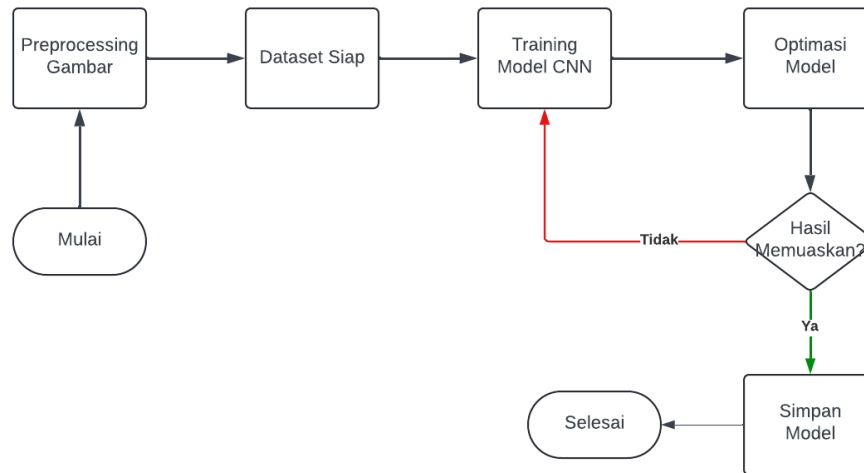
C. Pengumpulan Dataset

Dalam kasus ini dataset yang diperlukan adalah yang terdiri dalam jumlah besar gambar pantai dan pegunungan. Dataset ini harus cukup beragam untuk mencakup berbagai kondisi pencahayaan, sudut pengambilan gambar, dan elemen-elemen alam lainnya yang mungkin terdapat dalam foto. Setelah dataset dikumpulkan, langkah berikutnya adalah melakukan preprocessing pada gambar-gambar tersebut. *Preprocessing* ini meliputi perubahan ukuran gambar agar konsisten, normalisasi *pixel*, dan augmentasi data untuk meningkatkan jumlah data latih dan variasinya. Pengumpulan dataset menggunakan *tools* ekstensi Google Chrome bernama "Image downloader - Imageye". Penelitian ini menggunakan dataset yang terdiri dari 1420 gambar, yang terbagi sama rata antara foto pantai dan pegunungan (masing-masing 710 foto). Dataset ini dibagi menjadi data pelatihan dan data pengujian dengan rasio 80:20, sehingga terdapat 1136 gambar untuk *train* dan 284 gambar untuk test.

D. Preprocessing

Preprocessing adalah salah satu tahap pengolahan data gambar yang telah dikumpulkan sebelumnya untuk dilakukan penyesuaian. Pada penelitian ini terdapat beberapa tahap *preprocessing*, yaitu: mengubah ukuran, mengubah rentang nilai piksel, transformasi gambar, memperbesar gambar, dan mengubah orientasi gambar. Proses preprocessing bertujuan untuk membuat data gambar menjadi lebih seragam dan sesuai dengan format yang dibutuhkan oleh model CNN. Setelah melalui *preprocessing*, gambar-gambar tersebut sudah siap untuk dijadikan sebagai dataset pelatihan.

E. Training Model



Gambar 3. Rancangan Training Model

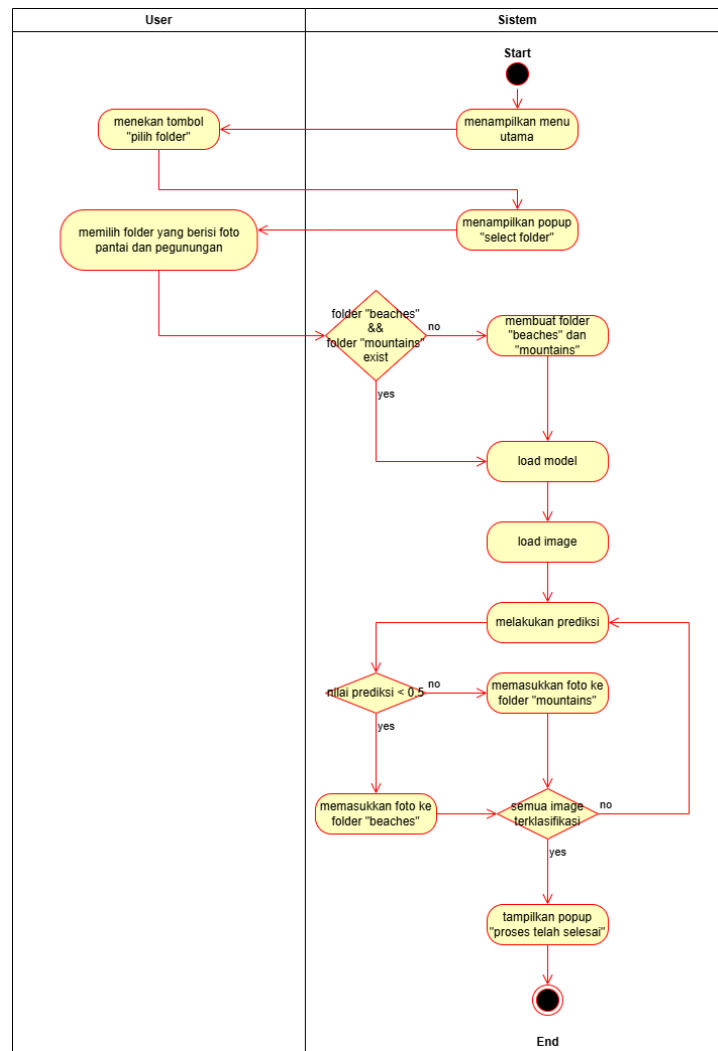
Dataset yang telah melalui tahap *preprocessing* dilatih menggunakan model CNN. Setelah dilatih, model akan dilakukan optimasi untuk meningkatkan performa model. Setelah model dioptimasi, perlu dilakukan mengevaluasi performanya. Jika hasil evaluasi belum memuaskan, maka dapat kembali ke tahap *training* atau optimasi untuk melakukan perbaikan. Jika hasil evaluasi sudah memuaskan, model yang telah dilatih akan disimpan dalam format h5.

F. Perancangan Perangkat Lunak

Perangkat lunak merupakan istilah yang digunakan untuk menggambarkan data yang disimpan dan diformat secara digital, termasuk program komputer, dokumentasi, dan informasi lain yang dapat dibaca dan diakses oleh komputer [7]. Sebelum dibuatnya perangkat lunak, diperlukan sebuah rancangannya terlebih dahulu agar program terstruktur dengan baik dan memenuhi kebutuhan yang diinginkan. Berikut adalah perancangan perangkat lunak:

1) Activity Diagram

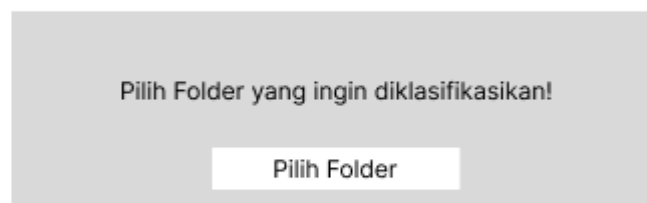
Activity Diagram merupakan salah satu jenis diagram *Unified Modelling Language* (UML) yang mengilustrasikan alur kerja atau aktivitas dari suatu sistem, proses bisnis, atau menu yang terdapat dalam perangkat lunak [8]. Gambar 4. menunjukkan proses pada saat program dimulai maka sistem akan menampilkan menu utama kepada user, lalu jika user menekan tombol “pilih folder” akan memerintahkan sistem untuk menampilkan popup “pilih folder” yang dimana popup tersebut memerintahkan *user* untuk memilih folder yang berisi foto pantai dan pegunungan. Setelah *user* memilih folder yang diperintahkan, maka sistem akan mengecek apakah dalam folder tersebut terdapat *subfolder* “*beaches*” dan *subfolder* “*mountains*”. Jika tidak ada maka sistem akan membuatkan kedua *subfolder* tersebut, namun jika kedua *subfolder* tersebut ada maka sistem akan *load* model, kemudian *load* gambar yang berada pada folder yang telah dipilih *user*, kemudian melakukan prediksi terhadap foto tersebut. Jika nilai prediksi kurang dari 0.5 maka masukkan foto tersebut dalam *subfolder* “*beaches*”, namun jika nilai prediksi lebih dari 0.5 maka masukkan foto dalam *subfolder* “*mountains*”. Proses tersebut dilakukan berulang - ulang sampai foto terakhir terklasifikasi, dan jika foto terakhir sudah terklasifikasi maka akan menampilkan popup “proses telah selesai”.



Gambar 4. Activity Diagram

2) Perancangan Graphical User Interface (GUI)

Graphical User Interface (GUI) adalah cara utama bagi pengguna untuk berinteraksi dengan perangkat guna memilih berbagai jenis tindakan. Elemen-elemen yang berinteraksi dengan pengguna, seperti tombol dan ikon, disebut objek. [9].



Gambar 5. Tampilan GUI pemilihan folder

Gambar 5 menunjukkan pesan instruksi untuk memilih folder yang berisi Pemandangan Pantai dan Pegunungan kemudian terdapat tombol “Pilih Folder” yang jika ditekan akan menampilkan popup “select folder”.

3. HASIL DAN PEMBAHASAN

A. Hasil Pengumpulan Dataset

Hasil dari pengumpulan dataset adalah berupa foto pemandangan pantai dan pegunungan yang diperoleh dari berbagai sumber. Total terdapat 1420 foto pemandangan, dengan jumlah yang sama rata yaitu 710 foto pantai dan 710 foto pegunungan. Kemudian dibuat folder bernama “dataset” yang

didalamnya terdapat *subfolder* “train” dan “test”. Setelah itu, sebanyak 568 foto pantai dan pegunungan dimasukkan ke dalam *subfolder* “train”, dan sisanya (142 foto pantai dan pegunungan) dimasukkan ke dalam *folder* “test”.

B. Hasil Preprocessing Dataset

Setelah dataset terkumpul, dilakukan preprocessing pada foto yang terdapat dalam folder “train” sebagai berikut :

1) *Resize*

Resize digunakan untuk mengubah ukuran citra menjadi ukuran yang diinginkan, berguna untuk mengurangi beban proses pada saat pelatihan data citra dan meningkatkan kualitas citra. Dalam data *train* nantinya akan di *resize* menjadi 150 x 150 piksel.

2) *Rescale*

Rescale adalah proses mengubah rentang nilai piksel dari gambar untuk membuatnya lebih sesuai untuk pelatihan model. Gambar digital biasanya memiliki nilai piksel dalam rentang 0-255, yang merepresentasikan intensitas warna. Untuk memfasilitasi pelatihan model, nilai-nilai ini sering kali diubah (di-*rescale*) ke rentang yang lebih kecil, biasanya 0-1 atau -1 sampai 1. Dalam data *train* menggunakan $rescale=1./255$, artinya kita mengubah setiap nilai piksel dari 0-255 menjadi 0-1, yaitu dengan membagi setiap nilai piksel dengan 255.

3) *Shear Range*

Shear Range adalah transformasi geometris yang memindahkan satu bagian dari gambar dalam satu arah sementara bagian lain dari gambar tetap dalam posisi yang sama. Dalam data *train* akan menggunakan $shear_range=0.3$ berarti gambar akan mengalami transformasi shear dengan sudut maksimum sebesar 0.3 radian (sekitar 17.2 derajat).

4) *Zoom Range*

Zoom range memiliki fungsi untuk memanipulasi gambar agar memiliki ukuran yang berbeda. Dalam data *train* menggunakan $zoom_range = 0.2$ yang artinya gambar dapat diperbesar atau diperkecil hingga 20% dari ukuran aslinya.

5) *Horizontal Flip*

Horizontal flip memiliki fungsi untuk memanipulasi gambar agar memiliki orientasi yang berbeda. Dalam hal ini, gambar dibalik secara *horizontal*, artinya baris dan kolom di balik ke arah kanan.

C. Hasil Pelatihan Model

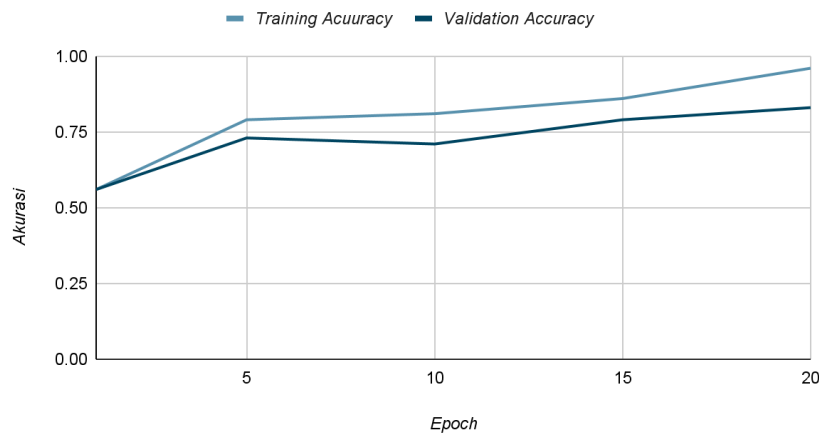
Setelah proses *preprocessing* selesai, dilakukanlah pelatihan. Pada tahap ini, model CNN akan mempelajari fitur-fitur penting dari fotopantai dan pegunungan melalui serangkaian lapisan konvolusi, *pooling*, dan *fully connected layer*. Selama proses ini, parameter-parameter model dioptimalkan menggunakan algoritma backpropagation dan gradient descent untuk mengurangi adanya kesalahan prediksi. Hasil *training* model yang dilakukan sebanyak 20 *epoch* dengan *batch size* 32 didapatkan hasil akhir berikut.

```

35/35 ----- 1s 30ms/step - accuracy: 0.6562 - loss: 0.7337 - val_accuracy: 0.6071 - val_loss: 0.7365
Epoch 3/20
35/35 ----- 51s 1s/step - accuracy: 0.7563 - loss: 0.5176 - val_accuracy: 0.7227 - val_loss: 0.5478
Epoch 4/20
35/35 ----- 1s 21ms/step - accuracy: 0.8125 - loss: 0.3676 - val_accuracy: 0.7143 - val_loss: 0.6623
Epoch 5/20
35/35 ----- 51s 1s/step - accuracy: 0.7955 - loss: 0.4437 - val_accuracy: 0.7383 - val_loss: 0.5396
Epoch 6/20
35/35 ----- 1s 21ms/step - accuracy: 0.7500 - loss: 0.4507 - val_accuracy: 0.6786 - val_loss: 0.6013
Epoch 7/20
35/35 ----- 50s 1s/step - accuracy: 0.8049 - loss: 0.4261 - val_accuracy: 0.7617 - val_loss: 0.5246
Epoch 8/20
35/35 ----- 1s 20ms/step - accuracy: 0.7812 - loss: 0.4310 - val_accuracy: 0.6429 - val_loss: 0.5686
Epoch 9/20
35/35 ----- 49s 1s/step - accuracy: 0.8202 - loss: 0.4097 - val_accuracy: 0.7617 - val_loss: 0.5322
Epoch 10/20
35/35 ----- 1s 18ms/step - accuracy: 0.8125 - loss: 0.3683 - val_accuracy: 0.7143 - val_loss: 0.4428
Epoch 11/20
35/35 ----- 49s 1s/step - accuracy: 0.8306 - loss: 0.4126 - val_accuracy: 0.7148 - val_loss: 0.6307
Epoch 12/20
35/35 ----- 1s 20ms/step - accuracy: 0.8750 - loss: 0.3283 - val_accuracy: 0.7500 - val_loss: 0.5057
Epoch 13/20
35/35 ----- 48s 1s/step - accuracy: 0.8418 - loss: 0.3571 - val_accuracy: 0.7188 - val_loss: 0.5986
Epoch 14/20
35/35 ----- 1s 26ms/step - accuracy: 0.8750 - loss: 0.2953 - val_accuracy: 0.7500 - val_loss: 0.5954
...
Epoch 19/20
35/35 ----- 50s 1s/step - accuracy: 0.8537 - loss: 0.3243 - val_accuracy: 0.8359 - val_loss: 0.3778
Epoch 20/20
35/35 ----- 1s 22ms/step - accuracy: 0.9688 - loss: 0.2484 - val_accuracy: 0.9286 - val_loss: 0.2523
    
```

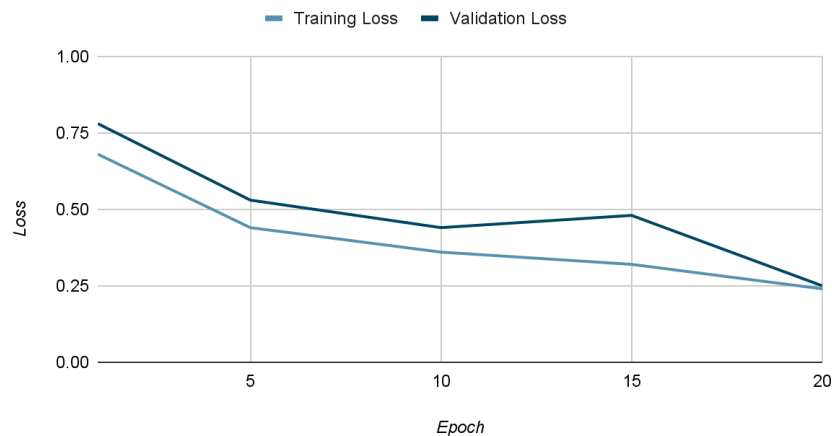
Gambar 6. Hasil Matriks pada Proses *Training Model*

Pada Gambar 6. menunjukkan hasil akhir *training* menggunakan dataset mendapat *accuracy* sebesar 96%, *loss* sebesar 24%, *val accuracy* sebesar 92%, dan *val_loss* sebesar 25%. Untuk lebih jelasnya bisa dilihat pada grafik *Accuracy* dan *Loss* berikut :



Gambar 7. Grafik Perbandingan *Accuracy* pada Proses *Training*

Pada Gambar 7. dapat diamati bahwa akurasi model meningkat seiring dengan bertambahnya jumlah *epoch*. Pada awalnya, akurasi meningkat dengan cepat dan kemudian cenderung stabil. Hal ini menunjukkan bahwa model belajar dengan baik dari data training dan menyesuaikan parameter-parameter internalnya untuk meminimalkan kesalahan. Sedangkan pada *Validation Accuracy* awalnya meningkat, tetapi kemudian fluktuasi sedikit sebelum akhirnya meningkat secara konsisten. Perbedaan kecil antara *training accuracy* dan *validation accuracy* menunjukkan bahwa model tidak mengalami *overfitting* dan memiliki generalisasi yang baik terhadap data baru.

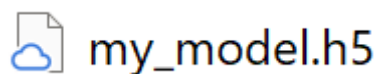


Gambar 8. Grafik Perbandingan fungsi Loss pada Proses Training

Pada Gambar 8. grafik menunjukkan *loss* model pada data *training* dan data *validation*. Terlihat bahwa *loss* pada data *training* berkurang secara konsisten seiring dengan bertambahnya jumlah *epoch*. Ini menunjukkan bahwa model semakin baik dalam memprediksi data *training* dengan kesalahan yang lebih kecil. *Loss* pada data *validation* juga menurun secara konsisten, dengan sedikit fluktuasi di tengah sebelum akhirnya menurun lagi. Penurunan ini menunjukkan bahwa model tidak hanya belajar dari data *training* tetapi juga mampu melakukan prediksi yang baik pada data baru yang tidak terlihat sebelumnya.

Kedua grafik menunjukkan konsistensi antara *training* dan *validation*, baik dari segi *accuracy* maupun *loss*. Ini adalah indikasi bahwa model tidak mengalami *overfitting* dan *underfitting* secara signifikan. Grafik menunjukkan bahwa model memiliki kemampuan generalisasi yang baik, yang berarti performa model pada data yang belum pernah dilihat hampir sama baiknya dengan data yang telah dilatih. Dengan akurasi yang meningkat dan *loss* yang menurun, dapat disimpulkan bahwa *training* model secara keseluruhan bekerja dengan baik dan mampu menangkap pola dari data dengan efektif. Ini adalah kondisi yang diharapkan dalam proses *training* model machine learning, di mana tujuan akhirnya adalah untuk mendapatkan model yang tidak hanya akurat pada data *training* tetapi juga mampu memprediksi data baru dengan baik.

Setelah dilakukan *training*, dan mendapatkan hasil akurasi yang memuaskan, maka dari hasil *training* tersebut bisa kita dapatkan model yang nantinya akan disimpan dalam bentuk file berformat *h5*. Berikut adalah hasil modelnya :



Gambar 9. Hasil Model

Pada gambar 9. menunjukkan model berformat *h5* yang nantinya akan digunakan untuk memprediksi foto pemandangan pantai dan pegunungan.

D. Hasil Perancangan Perangkat Lunak

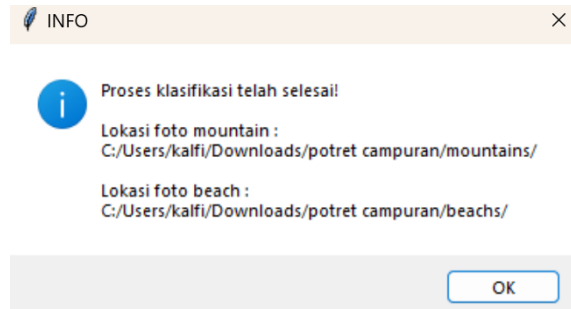
Berdasarkan *activity diagram* dan perancangan GUI yang telah dibuat, dihasilkan sebuah perangkat lunak berbasis desktop seperti gambar berikut ini :



Gambar 10. Hasil Perancangan Perangkat Lunak

Pada gambar 10. menunjukkan main menu dari perangkat lunak yang telah dirancang berdasarkan perancangan GUI. Sedangkan untuk logika dari perangkat lunaknya mengikuti diagram *activity* yang telah dibuat pada gambar 4. logika tersebut yaitu ketika tombol “pilih folder” ditekan, sebuah *popup*

“select folder” akan muncul, yang memungkinkan pengguna untuk memilih folder yang berisi foto pemandangan pantai dan pegunungan. Setelah pengguna memilih *folder* yang diinginkan, sistem memuat model (*file* berformat h5) yang telah disimpan sebelumnya. Lalu sistem akan melakukan klasifikasi hingga selesai dan jika selesai akan menampilkan *popup* sebagai berikut :

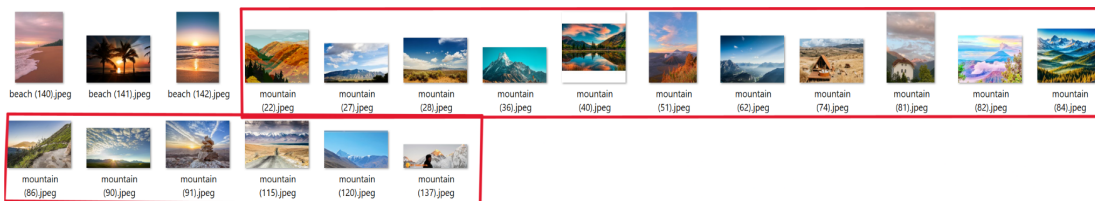


Gambar 11. *Popup* Info

Pada gambar 11. menunjukkan bahwa proses klasifikasi selesai dan semua hasil klasifikasi di simpan dalam *subfolder* “mountains” atau “beachs” tergantung dari hasil klasifikasinya.

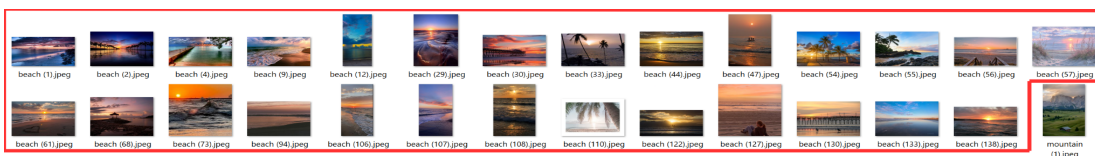
E. Hasil Klasifikasi

Setelah perangkat lunak dikembangkan, dilakukanlah testing yaitu menggunakan 284 foto pemandangan yang merupakan gabungan antara 142 foto pantai dan 142 foto pegunungan. Semua foto tersebut berada dalam satu folder yang bernama "campuran". Setelah program dijalankan, secara otomatis di dalam folder "campuran" dibuat dua folder baru yang masing-masing bernama "*beachs*" dan "*mountains*". Selanjutnya, semua foto tersebut akan diklasifikasi oleh program. Jika foto terklasifikasi sebagai pantai, maka akan dimasukkan ke dalam *folder* "*beachs*", dan jika foto terklasifikasi sebagai pegunungan, maka akan masuk ke dalam folder "*mountains*".



Gambar 12. Isi *Folder* Beachs

Pada Gambar 12, hasil klasifikasi menunjukkan bahwa *folder* “beachs” mengandung 115 gambar pantai yang sesuai dengan kategori yang seharusnya, tetapi juga terdapat 17 gambar pegunungan yang dimasukkan secara salah. Ini menunjukkan bahwa meskipun label *folder* “beachs” mengindikasikan bahwa sebagian besar foto di dalamnya adalah foto pantai, ada kesalahan klasifikasi yang menyebabkan foto-foto pegunungan juga masuk ke dalam folder tersebut. Kesalahan ini menunjukkan bahwa ada masalah dengan proses pengkategorian otomatis yang dilakukan oleh program. Kesalahan klasifikasi ini bisa disebabkan oleh berbagai faktor, seperti adanya fitur visual yang mirip antara foto pantai dan foto pegunungan, atau karena algoritma klasifikasi yang digunakan tidak cukup akurat dalam membedakan kedua kategori tersebut.



Gambar 13. Isi *Folder* Mountains.

Selanjutnya, pada Gambar 13, hasil klasifikasi untuk folder “mountains” menunjukkan bahwa terdapat total 125 foto pegunungan, namun diantara foto-foto tersebut, terdapat 27 foto pantai yang seharusnya tidak berada di sana. Hal ini menunjukkan bahwa dari total 142 foto pantai, sebanyak 27 foto pantai gagal terdeteksi dengan benar sebagai foto pantai dan malah terklasifikasi sebagai foto

pegunungan. Demikian pula, dari total 142 foto pegunungan, terdapat 17 foto yang seharusnya dikategorikan sebagai foto pegunungan tetapi salah teridentifikasi sebagai foto pantai.

Tabel 1. Hasil Klasifikasi Gambar

Jenis Gambar	Total Gambar	Berhasil	Gagal
Pantai	142	115	27
Pegunungan	142	125	17

Secara keseluruhan, dari 284 foto yang diolah oleh sistem klasifikasi, sebanyak 240 foto berhasil teridentifikasi dengan benar sesuai dengan kategorinya, sedangkan 44 foto lainnya gagal teridentifikasi. Kesalahan klasifikasi ini menunjukkan adanya tantangan dalam membedakan antara foto pantai dan pegunungan dengan tepat, meskipun mayoritas klasifikasi yang dilakukan oleh sistem sudah benar. Tantangan ini bisa jadi disebabkan oleh beberapa faktor, termasuk kualitas gambar, pencahayaan, sudut pengambilan gambar, serta kesamaan fitur visual antara foto pantai dan pegunungan.

4. KESIMPULAN

Penelitian ini mendapatkan hasil yang memuaskan dalam pengembangan model *Convolutional Neural Network* (CNN) untuk klasifikasi gambar pemandangan pantai dan pegunungan. Dataset yang digunakan terdiri dari 1420 gambar yang dibagi secara proporsional antara gambar pantai dan pegunungan. Pembagian data sebesar 80% dialokasikan untuk pelatihan dan 20% dialokasikan untuk pengujian. Model CNN yang dibangun terdiri dari sembilan lapisan—termasuk lapisan konvolusi, *pooling*, *flatten*, dan *fully connected* dan dilatih selama 20 *epoch*. Hasil pelatihan menunjukkan akurasi 96% dan validasi 92%, dan model disimpan sebagai *my_model.h5*. Hasil perangkat lunak juga berjalan sesuai yang diinginkan, program dapat memilih gambar yang ingin diklasifikasikan dan memisahkannya di folder yang berbeda. Hasil klasifikasi dari perangkat lunak menunjukkan bahwa dari 142 gambar pantai yang berhasil terklasifikasi sebagai pantai sebanyak 115 gambar, dan dari 142 gambar pegunungan yang terklasifikasi sebagai gambar pegunungan sebanyak 125 gambar. Hal tersebut menunjukkan bahwa model yang telah dibangun sesuai dengan hasil yang diharapkan.

DAFTAR PUSTAKA

- [1] Hermawan, E. (2021). Klasifikasi Pengenalan Wajah Menggunakan Masker atau Tidak Dengan Mengimplementasikan Metode CNN (Convolutional Neural Network). *Jurnal Industri Kreatif Dan Informatika Series (JIKIS)*, 1(1), 33–43.
- [2] Mira, M., Sembiring, I., & Purnomo, H. D. (2022). Implementasi Transfer Learning Pada Algoritma Convolutional Neural Network untuk Mengklasifikasikan Image Objek Wisata. *Building of Informatics, Technology and Science (BITS)*, 4(1). <https://doi.org/10.47065/bits.v4i1.1764>
- [3] Kurniawan, M. H. (2024). APLIKASI IDENTIFIKASI KENDARAAN KARYAWAN DENGAN METODE CONVOLUTIONAL NEURAL NETWORK. *Scientica: Jurnal Ilmiah Sains Dan Teknologi*, 2(2), 251–256.
- [4] ANHAR, A., & PUTRA, R. A. (2023). Perancangan dan Implementasi Self-Checkout System pada Toko Ritel menggunakan Convolutional Neural Network (CNN). *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, 11(2), 466. <https://doi.org/10.26760/elkomika.v11i2.466>.
- [5] Setiawan, T. (2019). Implementasi Convolutional Neural Network Untuk Pengenalan Warna Kendaraan. *Doctoral Dissertation, University of Technology Yogyakarta*.
- [6] Rena, P. N. (2019). Penerapan Metode Convolutional Neural Network Pada Pendeteksi Gambar Notasi Balok. *UIN Syarif Hidayatullah Jakarta*.
- [7] Hedro Dwi Cahyono, W., Adiyasa Putra, R., Fikri Albari, M., Yanuar, M., Apriliyadi, A., Nakita Pratama, A., Mechael Beda, B., Pratama Setiya, U., & Paladan Sambolayuk, W. (2023). BIMBINGAN PENGENALAN PERANGKAT LUNAK KOMPUTER KEPADA SISWA-SISWI SMP DJOJOREDJO. *Abdi Jurnal Publikasi*, 1(6), 510–514. <https://jurnal.portalpublikasi.id/index.php/AJP/index510>

- [8] Musthofa, N., & Adiguna, M. A. (2022). Perancangan Aplikasi E-Commerce Spare-Part Komputer Berbasis Web Menggunakan CodeIgniter Pada Dhamar Putra Ccomputer Kota Tangerang. *OKTAL: Jurnal Ilmu Komputer Dan Science*, 1(03). <https://journal.mediapublikasi.id/index.php/oktal>.
- [9] Sistem, J., Ayu, D., & Utami, B. (2021). Perancangan Sistem Login Pada Aplikasi Berbasis GUI Menggunakan QTDesigner Python. *Jurnal SIMADA (Sistem Informasi Dan Manajemen Basis Data)*, 4(2).
- [10] ALAMIN, D. (2024). *IMPLEMENTASI METODE CONVOLUTIONAL NEURAL NETWORK PADA KLASIFIKASI CITRA GUNUNG MERAPI* (Doctoral dissertation, UPN" Veteran" Yogyakarta).
- [11] Arjun, A. (2024). Klasifikasi Citra Pada Tingkat Kematangan Buah Pisang Menggunakan Algoritma Deep Learning. *Jurnal Ekonomi Manajemen Sistem Informasi*, 5(3), 203–208. <https://doi.org/10.31933/jemsi.v5i3>
- [12] Arrofiqoh, E. N., & Harintaka, H. (2018). IMPLEMENTASI METODE CONVOLUTIONAL NEURAL NETWORK UNTUK KLASIFIKASI TANAMAN PADA CITRA RESOLUSI TINGGI. *GEOMATIKA*, 24(2), 61. <https://doi.org/10.24895/JIG.2018.24-2.810>
- [13] Arvio, Y., Kusuma, D. T., Sangadji, I. B., & Dewantara, E. K. (2024). Penerapan Metode Convolution Neural Network (CNN) Dalam Proses Pengolahan Citra Untuk Mendeteksi Cacat Produksi Pada Produk Masker. *Faktor Exacta*, 16(4). <https://doi.org/10.30998/faktorexacta.v16i4.20073>
- [14] Charli, F., Syaputra, H., Akbar, M., Sauda, S., & Panjaitan, F. (2020). Implementasi Metode Faster Region Convolutional Neural Network (Faster R-CNN) Untuk Pengenalan Jenis Burung Lovebird. *Journal of Information Technology Ampera*, 1(3), 185–197. <https://doi.org/10.51519/journalita.volume1.issue3.year2020.page185-197>
- [15] Reza Fahcuroji, A., Yunita Wijaya, M., Fauziah, I., Sains dan Teknologi, F., Syarif Hidayatullah Jakarta Jl Ir Juanda No, U. H., Ciputat Tim, K., & Tangerang Selatan, K. (2024). IMPLEMENTASI ALGORITMA CNN MOBILENET UNTUK KLASIFIKASI GAMBAR SAMPAH DI BANK SAMPAH. *Jurnal Pengembangan Riset Dan Observasi Sistem Komputer*, 11(1), 45–51.
- [16] Jumaryadi, Y., Muhammad Ihsan, A., & Priambodo, B. (2023). KLIK: Kajian Ilmiah Informatika dan Komputer Klasifikasi Jenis Buah-Buahan Menggunakan Citra Digital Dengan Metode Convolutional Neural Networks. *Media Online*, 4(3), 1737–1746. <https://doi.org/10.30865/klik.v4i3.1421>
- [17] Lia Farokhah. (2020). Implementasi Convolutional Neural Network untuk Klasifikasi Variasi Intensitas Emosi pada Dynamic Image Sequence. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 4(6). <https://doi.org/10.29207/resti.v4i6.2644>
- [18] Maulana, R., Dwi, R., Putri, Z., Fitriani, S., Sihaloho, M., & Mulyana, S. (2023). Implementasi Algoritma Convolutional Neural Network Dalam Mengklasifikasi Jenis Burung. *Journal of Creative Student Research (JCSR)*, 1(6), 221–231. <https://doi.org/10.55606/jcsrpolitama.v1i6.2966>
- [19] Maha, V., Salawazo, P., Putra, D., Gea, J., Gea, R. F., & Azmi, F. (2019). Terakreditasi DIKTI No.SK 1,2,3 Mahasiswa Teknik Informatika. *Indonesia Jalan Sekip Sikambang*, 3(1), 74–79.
- [20] Sari, A. P., Suzuki, H., Kitajima, T., Yasuno, T., & Prasetya, D. A. (2020). PREDICTION MODEL OF WIND SPEED AND DIRECTION USING DEEP NEURAL NETWORK. *JEEMECS (Journal of Electrical Engineering, Mechatronic and Computer Science)*, 3(1). <https://doi.org/10.26905/jeemeecs.v3i1.3946>