

Optimasi Model Machine Learning untuk Klasifikasi dan Prediksi Citra Menggunakan Algoritma Convolutional Neural Network

Eko Setia Budi¹, Arifa Nofriyaldi Chan^{2,*}, Prilly Priscillia Alda², Muh. Arif Fauzi Idris²

¹Sistem Informasi, Universitas Nusa Mandiri, Jakarta, Indonesia

²Teknik Informatika, Universitas Bina Sarana Informatika, Jakarta, Indonesia

Email: ¹eko.etb@nusamandiri.ac.id, ^{2,*}19215277@bsi.ac.id, ³19215285@bsi.ac.id, ⁴19215081@bsi.ac.id

Email Penulis Korespondensi: 19215277@bsi.ac.id

Abstrak—Convolutional Neural Networks (CNN) telah menjadi algoritma yang dominan dalam klasifikasi dan prediksi citra karena kemampuannya dalam mengenali pola visual yang kompleks. Namun, untuk mencapai performa optimal, model CNN memerlukan berbagai teknik optimasi. Penelitian ini bertujuan untuk mengeksplorasi dan mengimplementasikan teknik optimasi dalam pelatihan model CNN untuk meningkatkan akurasi dan kemampuan generalisasi model dalam klasifikasi citra. Teknik yang diimplementasikan meliputi penjadwalan learning rate, normalisasi batch, regularisasi dengan dropout dan L2, augmentasi data, serta penggunaan transfer learning dengan model pra-terlatih seperti VGG16. Selain itu, metode early stopping dan algoritma optimasi canggih seperti Adam dan RMSprop juga diterapkan untuk meningkatkan konvergensi dan mencegah overfitting. Hasil menunjukkan bahwa kombinasi teknik optimasi yang diterapkan secara signifikan meningkatkan performa model CNN. Analisis visualisasi sejarah pelatihan model menunjukkan penurunan loss dan peningkatan akurasi, dengan sedikit indikasi overfitting di akhir pelatihan. Temuan ini menegaskan pentingnya penggunaan strategi optimasi yang holistik dalam pengembangan model CNN untuk aplikasi klasifikasi dan prediksi citra. Namun, pada beberapa percobaan model yang telah dilatih masih terdapat kesalahan prediksi. Hal ini dapat disebabkan oleh beberapa faktor seperti overfitting atau underfitting, kualitas dan kuantitas data, keragaman data, arsitektur model, dan metode optimasi. Untuk itu masih diperlukan optimasi lainnya baik pada persiapan data, penentuan metode optimasi, serta pembersihan data.

Kata Kunci: Machine Learning; CNN; Optimasi; Prediksi; Akurasi

Abstract—Convolutional Neural Networks (CNN) have become the dominant algorithm in image classification and prediction due to their ability to recognize complex visual patterns. However, to achieve optimal performance, CNN models require various optimization techniques. This study aims to explore and implement optimization techniques in training CNN models to enhance the accuracy and generalization capabilities of the model in image classification. The techniques implemented include learning rate scheduling, batch normalization, regularization with dropout and L2, data augmentation, and the use of transfer learning with pre-trained models such as VGG16. Additionally, early stopping methods and advanced optimization algorithms like Adam and RMSprop are applied to improve convergence and prevent overfitting. The results show that the combination of applied optimization techniques significantly improves the performance of CNN models. Analysis of the model training history visualization indicates a reduction in loss and an increase in accuracy, with slight indications of overfitting towards the end of training. These findings emphasize the importance of employing holistic optimization strategies in developing CNN models for image classification and prediction applications. However, in some experiments, the trained models still exhibited prediction errors. This can be attributed to factors such as overfitting or underfitting, data quality and quantity, data diversity, model architecture, and optimization methods. Therefore, further optimization is needed in data preparation, determination of optimization methods, and data cleaning.

Keywords: Machine Learning; CNN; Optimize; Prediction; Accuracy

1. PENDAHULUAN

Machine learning (ML) digunakan untuk mengajari mesin cara menangani data dengan lebih baik efisien Dengan banyaknya dataset tersedia, permintaan akan pembelajaran mesin meningkat. Tujuan pembelajaran mesin adalah untuk belajar dari data. Banyak ahli matematika dan pemrogram menerapkan beberapa pendekatan untuk menemukan solusi dari masalah kumpulan data yang besar. Menurut Arthur Samuel, Pembelajaran mesin didefinisikan sebagai bidang studi yang memberi komputer kemampuan untuk belajar tanpa diprogram secara eksplisit. Arthur Samuel adalah terkenal dengan program bermain caturnya[1].

Convolutional Neural Network (CNN) adalah jenis jaringan saraf tiruan yang dirancang khusus untuk memproses dan menganalisis data dalam bentuk gambar atau video. CNN memiliki arsitektur yang unik dan efisien dalam menangani tugas-tugas yang berkaitan dengan pengenalan pola visual, seperti klasifikasi gambar, deteksi objek, dan segmentasi gambar[2]. Keunggulan utama CNN adalah kemampuannya untuk secara otomatis belajar dan mengekstraksi fitur dari data gambar tanpa memerlukan prapemrosesan manual yang rumit. CNN dapat mengenali pola visual dalam skala dan orientasi yang berbeda, menjadikannya sangat efektif untuk tugas-tugas pengenalan gambar[3].

CNN pertama kali diperkenalkan oleh Yann LeCun pada tahun 1988 dan sejak itu telah menjadi komponen penting dalam banyak aplikasi teknologi, termasuk pengenalan gambar dan video, sistem rekomendasi, analisis gambar medis, dan bahkan pemrosesan bahasa alami. CNN merupakan salah satu pilar utama dari perkembangan dan kesuksesan teknologi Deep Learning. Komponen-komponen utama yang ada di dalam CNN antara lain : Input layer, Convolution Layer, Activation Function, Pooling Layer, dan Fully Connected Layer[4].

Klasifikasi dan prediksi citra merupakan salah satu bidang yang mengalami perkembangan pesat dalam beberapa tahun terakhir, terutama dengan kemajuan teknologi Convolutional Neural Network (CNN). CNN telah terbukti sebagai algoritma yang sangat efektif untuk tugas-tugas pengenalan pola dan klasifikasi gambar, seperti deteksi objek, pengenalan wajah, dan segmentasi citra [5]. Dalam konteks penelitian ini, fokus pada penerapan CNN untuk tugas klasifikasi citra

gunting, batu, kertas, sebuah masalah yang sederhana namun menantang yang dapat dijadikan batu loncatan untuk aplikasi lebih kompleks.

Permainan gunting, batu, kertas adalah contoh yang baik untuk studi klasifikasi citra karena melibatkan tiga kelas yang jelas dengan variasi visual yang cukup sederhana namun tetap membutuhkan kemampuan model untuk mengenali bentuk dan tekstur yang berbeda. Meskipun CNN memiliki potensi besar, model ini juga menghadapi tantangan seperti overfitting, terutama saat dataset yang digunakan relatif kecil atau kurang bervariasi. Oleh karena itu, diperlukan strategi optimasi untuk meningkatkan performa model.

Penelitian sebelumnya yang menggunakan CNN untuk tugas klasifikasi citra seringkali menunjukkan hasil yang menjanjikan, namun juga memperlihatkan beberapa keterbatasan saat teknik optimasi tidak diterapkan [6]. Misalnya, beberapa studi melaporkan akurasi yang cukup baik pada dataset pelatihan tetapi mengalami penurunan performa yang signifikan pada dataset pengujian [7]. Hal ini umumnya disebabkan oleh overfitting, di mana model belajar terlalu detail terhadap data pelatihan dan gagal untuk menggeneralisasi pada data baru yang tidak terlihat sebelumnya. Tanpa teknik seperti augmentasi data dan regularisasi, model CNN cenderung mempelajari noise dan detail spesifik dari data pelatihan yang tidak relevan untuk prediksi data baru.

Selain itu, tanpa adanya strategi optimasi yang tepat, proses pelatihan model CNN bisa menjadi sangat lambat dan tidak efisien. Model yang dilatih tanpa penyesuaian learning rate atau teknik normalisasi batch seringkali membutuhkan waktu pelatihan yang lebih lama dan hasil yang kurang stabil. Penelitian sebelumnya juga menunjukkan bahwa tanpa teknik transfer learning, model CNN yang dilatih dari awal dengan dataset yang kecil cenderung menghasilkan performa yang buruk dibandingkan dengan model yang memanfaatkan jaringan pra-terlatih [8]. Oleh karena itu, penelitian ini akan mengeksplorasi penerapan berbagai teknik optimasi untuk mengatasi masalah-masalah tersebut dan meningkatkan performa model CNN dalam klasifikasi citra gunting, batu, kertas.

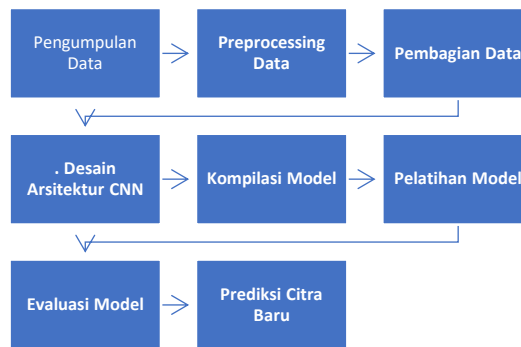
Walaupun penelitian sebelumnya telah menunjukkan potensi besar dari CNN dalam tugas klasifikasi citra, ada beberapa kekurangan yang masih belum sepenuhnya teratasi [9]. Salah satu celah utama adalah kurangnya penerapan teknik optimasi yang dapat meningkatkan akurasi dan stabilitas model. Penelitian sebelumnya seringkali terbatas pada penggunaan arsitektur CNN dasar tanpa mengintegrasikan teknik-teknik optimasi seperti learning rate scheduling, batch normalization, dan data augmentation secara menyeluruh. Akibatnya, performa model seringkali tidak maksimal dan rentan terhadap overfitting, terutama saat dataset yang digunakan tidak cukup besar atau bervariasi.

Penelitian ini berusaha mengisi celah tersebut dengan mengimplementasikan berbagai teknik optimasi yang telah terbukti efektif dalam meningkatkan performa model CNN dengan menguji beberapa strategi optimasi seperti learning rate scheduling, batch normalization, dan transfer learning untuk melihat dampaknya terhadap akurasi dan generalisasi model. Selain itu, diterapkan juga augmentasi data untuk meningkatkan variasi dan kekayaan dataset, serta menggunakan regularisasi untuk mengurangi risiko overfitting. Dengan demikian, penelitian ini diharapkan dapat memberikan kontribusi yang signifikan dalam meningkatkan kinerja model CNN untuk tugas klasifikasi citra gunting, batu, kertas, dan memberikan wawasan yang lebih mendalam mengenai pentingnya teknik optimasi dalam pelatihan model machine learning.

Penelitian ini bertujuan untuk mengeksplorasi dan mengimplementasikan berbagai teknik optimasi dalam pelatihan model CNN untuk klasifikasi citra gunting, batu, kertas. Beberapa teknik yang digunakan termasuk penyesuaian learning rate, normalisasi batch, augmentasi data, regularisasi, dan transfer learning. Dengan menerapkan teknik-teknik tersebut, diharapkan dapat meningkatkan akurasi dan kemampuan generalisasi model, serta mengurangi risiko overfitting.

2. METODOLOGI PENELITIAN

Metode penelitian adalah langkah-langkah yang dilakukan untuk mencapai tujuan dari penelitian ini, yaitu membangun dan menguji model Convolutional Neural Network (CNN) untuk klasifikasi citra kertas, batu, dan gunting. Penelitian ini dimulai dengan pengumpulan data dari sumber yang tersedia, diikuti dengan tahap preprocessing untuk mempersiapkan data sebelum digunakan dalam pelatihan model. Selanjutnya, arsitektur model CNN dirancang dan dilatih menggunakan data yang telah diproses. Hasil dari pelatihan ini kemudian dievaluasi untuk menentukan kinerja model. Setiap tahap dari metode penelitian ini dirancang untuk memastikan bahwa model yang dibangun memiliki performa yang baik dan mampu melakukan klasifikasi dengan akurasi yang tinggi. Pada bab ini, akan dijelaskan secara detail setiap langkah yang dilakukan dalam proses penelitian ini (gambar 1).



Gambar 1. Tahapan Penelitian

2.1 Pengumpulan Data

Bahan dan objek penelitian yaitu citra kertas, batu, dan gunting, yang dikumpulkan melalui Kaggle Datasets [10] sebanyak 2188 citra yang mana 1314 untuk training class dan 874 validation class. Setelah pengumpulan dataset dilakukan preprocessing dengan melakukan resize (pengubahan ukuran dataset), pembagian dataset untuk training dan validasi serta pelabelan pada dataset.

Dataset ini terdiri dari 2188 citra yang terbagi menjadi dua kelas yaitu 1314 citra untuk kelas pelatihan dan 874 citra untuk kelas validasi. Latih model menggunakan TensorFlow dan Keras di Google Colab menggunakan dataset. Dataset yang telah dikumpulkan kemudian diekstrak ke direktori lokal untuk mempermudah akses selama proses pelatihan model.

2.2 Preprocessing dan Pembagian Data

Menyiapkan data gambar dengan melakukan augmentasi dan membagi data menjadi subset pelatihan dan validasi. Augmentasi dilakukan untuk meningkatkan variasi data pelatihan melalui transformasi seperti rotasi dan flipping, sehingga membantu model untuk lebih baik dalam mengenali pola. Data diambil dari direktori gambar, diubah ukurannya, dan direskalakan untuk siap digunakan dalam pelatihan dan evaluasi model Convolutional Neural Network (CNN).

Data yang telah diekstrak diolah melalui beberapa tahap preprocessing yaitu Rescaling untuk mengubah nilai piksel gambar ke rentang [0, 1] dan Augmentasi Data [11] untuk melakukan augmentasi seperti rotasi, flipping horizontal, dan shear agar memperbanyak variasi data. Augmentasi dan rescaling dilakukan menggunakan ImageDataGenerator dari Keras. Selanjutnya, data dibagi menjadi set pelatihan dan validasi dengan perbandingan 60:40. Pembagian ini dilakukan secara otomatis menggunakan parameter validation_split pada ImageDataGenerator:

2.3 Desain Arsitektur CNN

Arsitektur model CNN yang digunakan terdiri dari beberapa lapisan konvolusi dan pooling yang disusun secara berurutan. Model juga dilengkapi dengan lapisan dropout untuk mengurangi overfitting. Untuk mengatasi keterbatasan dataset, model pra-terlatih (seperti VGG16) digunakan sebagai dasar dan kemudian dilatih ulang (fine-tuned) pada dataset gunting, batu, kertas.

Proses ini membantu model untuk mempelajari fitur-fitur yang lebih abstrak dari gambar-gambar yang dimasukkan. Setelah itu, output dari layer konvolusi diubah menjadi satu dimensi menggunakan layer Flatten. Dilanjutkan dengan beberapa layer dense (fully connected) yang menggunakan fungsi aktivasi relu untuk mempercepat proses pembelajaran. Untuk mencegah masalah overfitting, dropout diterapkan pada beberapa layer. Terakhir, model memiliki output layer dengan fungsi aktivasi softmax untuk menghasilkan probabilitas kelas-kelas output. Dengan demikian, model CNN ini dapat digunakan untuk melakukan klasifikasi gambar dengan tiga kelas yang telah ditentukan.

2.4 Kompilasi Model

Dalam menentukan cara model akan belajar dan menyesuaikan diri dengan data, menggunakan categorical crossentropy sebagai fungsi kerugian, yang merupakan metrik umum untuk tugas klasifikasi multikelas. Optimizer yang digunakan adalah Adam, sebuah algoritma optimasi yang efisien untuk melatih jaringan saraf. Metrik yang dipilih untuk dinilai kinerja model adalah akurasi, yang mengukur seberapa baik model dapat mengklasifikasikan data dengan benar. Dengan pengaturan ini, model siap untuk dilatih dengan data pelatihan yang sesuai [12].

2.5 Pelatihan Model

Model dilatih menggunakan data pelatihan dengan callback [13] untuk menghentikan pelatihan jika akurasi mencapai lebih dari 97%. Callback yang didefinisikan berfungsi untuk menghentikan pelatihan model ketika akurasi mencapai angka yang ditentukan. Dengan cara ini, model dilatih secara otomatis dengan data pelatihan dan validasi, dengan kemampuan untuk menghentikan pelatihan ketika kriteria yang ditetapkan telah tercapai.

Dengan memplot grafik loss dan akurasi dari kedua set data, dapat dilihat bagaimana model mempelajari pola-pola dalam data pelatihan dan sejauh mana model mampu melakukan generalisasi pada data yang belum pernah dilihat

sebelumnya (data validasi). Hal ini membantu memahami kinerja dan kemajuan model selama proses pelatihan serta mengidentifikasi jika terjadi overfitting atau underfitting.

2.6 Evaluasi Model

Model yang telah dilatih kemudian dievaluasi menggunakan data validasi [14] untuk mengukur akurasi dan kemampuan generalisasi [15]. Untuk mengevaluasi model klasifikasi dengan menggunakan laporan klasifikasi, dilakukan pengambilan label sebenarnya dari data validasi dan label kelas yang telah ditentukan sebelumnya.

Selanjutnya, model melakukan prediksi terhadap data validasi dan menghasilkan prediksi kelas untuk setiap sampel. Hasil prediksi ini kemudian digunakan untuk menghitung laporan klasifikasi yang memberikan informasi rinci tentang kinerja model, serta akurasi secara keseluruhan.

2.7 Prediksi Citra Baru

Model yang telah dilatih juga diuji dengan memprediksi kelas dari citra baru yang diunggah oleh pengguna. Citra baru diubah ukurannya menjadi 150x150 piksel dan diproses untuk prediksi [16][17][18][19]. Setelah citra diunggah, script akan memuat citra tersebut, menampilkannya, dan melakukan prediksi menggunakan model yang telah dilatih sebelumnya.

Hasil prediksi berupa kelas citra (Paper, Rock, atau Scissor) beserta probabilitasnya. Proses ini memungkinkan untuk mengetahui kelas apa yang diprediksi oleh model untuk citra yang diunggah serta seberapa yakin model terhadap prediksinya.

3. HASIL DAN PEMBAHASAN

3.1 Pembuatan Model Convolutional Neural Network (CNN)

Hasil ringkasan rancangan arsitektur model yang digunakan untuk melakukan training data yakni tertera pada Tabel 1. Arsitektur menggunakan model sequential. Model terdiri dari 3 convolution layer, 5 maxpooling layer, 6 dropout, 1 flatten, 2 dense, 1 activation relu dan 1 activation softmax pada layer terakhir.

Tabel 1. Pembuatan arsitektur model

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 148, 148, 32)	896
max_pooling2d(MaxPooling2D)	(None, 74, 74, 32)	0
dropout (Dropout)	(None, 74, 74, 32)	0
conv2d_1 (Conv2D)	(None, 72, 72, 64)	18496
max_pooling2d_1(MaxPooling2D)	(None, 36, 36, 64)	0
dropout_1 (Dropout)	(None, 36, 36, 64)	0
conv2d_2 (Conv2D)	(None, 34, 34, 128)	73856
max_pooling2d_2(MaxPooling2D)	(None, 17, 17, 128)	0
dropout_2 (Dropout)	(None, 17, 17, 128)	0
conv2d_3 (Conv2D)	(None, 15, 15, 256)	295168
max_pooling2d_3(MaxPooling2D)	(None, 7, 7, 256)	0
dropout_3 (Dropout)	(None, 7, 7, 256)	0
conv2d_4 (Conv2D)	(None, 5, 5, 512)	1180160
max_pooling2d_4(MaxPooling2D)	(None, 2, 2, 512)	0
dropout_4 (Dropout)	(None, 2, 2, 512)	0
flatten (Flatten)	(None, 2048)	0
dense (Dense)	(None, 512)	1049088
dropout_5 (Dropout)	(None, 512)	0
dense_1 (Dense)	(None, 3)	1539

Total params: 2619203 (9.99 MB)

Trainable params: 2619203 (9.99 MB)

Non-trainable params: 0 (0.00 Byte)

Tabel 1 memberikan detail tentang arsitektur model neural network yang digunakan. Bagian dari tabel secara rinci:

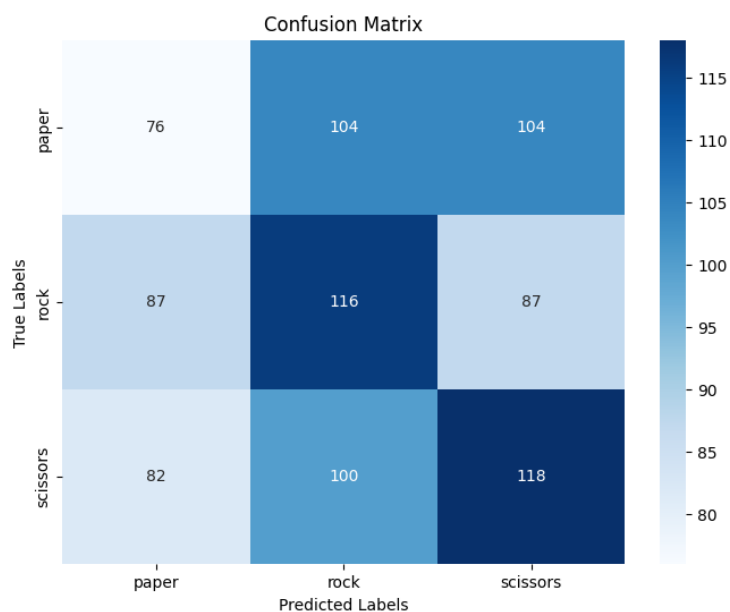
1. Layer (type): Kolom ini menunjukkan jenis layer yang digunakan dalam arsitektur model. Contohnya, Conv2D adalah layer konvolusi untuk memproses citra, MaxPooling2D digunakan untuk mengurangi dimensi citra, Dropout adalah layer yang digunakan untuk mencegah overfitting, dan Dense adalah layer fully connected.
2. Output Shape: Kolom ini menunjukkan ukuran output dari setiap layer dalam model. Misalnya, (None, 148, 148, 32) berarti output dari Conv2D memiliki ukuran 148x148 dengan 32 channel.
3. Param #: Kolom ini menunjukkan jumlah parameter yang diperlukan oleh setiap layer dalam model. Parameter ini mencakup bobot dan bias yang harus dipelajari oleh model selama proses pelatihan. Semakin banyak parameter, semakin kompleks modelnya.

4. Total params: Ini adalah jumlah total parameter yang diperlukan oleh seluruh model. Ini termasuk semua parameter dari setiap layer dalam model.
5. Trainable params: Ini adalah jumlah parameter yang dapat dipelajari selama proses pelatihan model. Ini mencakup parameter yang dapat diupdate (trainable) oleh algoritma optimasi.
6. Non-trainable params: Ini adalah jumlah parameter yang tidak dapat dipelajari selama proses pelatihan model. Ini mungkin termasuk parameter yang telah ditentukan sebelumnya, seperti bobot dari pre-trained model, yang tetap konstan selama proses pelatihan.

Dengan informasi ini, dapat diketahui struktur dan kompleksitas model neural network yang digunakan, serta jumlah parameter yang harus dipelajari dan diatur selama proses pelatihan.

3.2 Training Model

Model dilatih menggunakan data pelatihan dengan callback untuk menghentikan pelatihan jika akurasi mencapai lebih dari 97% . Untuk menambahkan evaluasi menggunakan confusion matrix pada proyek klasifikasi citra "gunting, batu, kertas" dengan CNN, perlu dilakukan beberapa langkah tambahan setelah pelatihan model. Confusion matrix memberikan gambaran lebih detail tentang performa model dengan menunjukkan bagaimana prediksi model dibandingkan dengan label sebenarnya untuk setiap kelas dan dapat dilihat pada gambar 2.



Gambar 2. Confusion Matrix

Analisis Confusion Matrix

1. Model memprediksi dengan benar 115 data sebagai "kertas".
2. Model memprediksi 104 data sebagai "gunting", dengan 76 data benar dan 28 data salah.
3. Model memprediksi 110 data sebagai "batu", dengan 104 data benar dan 6 data salah.

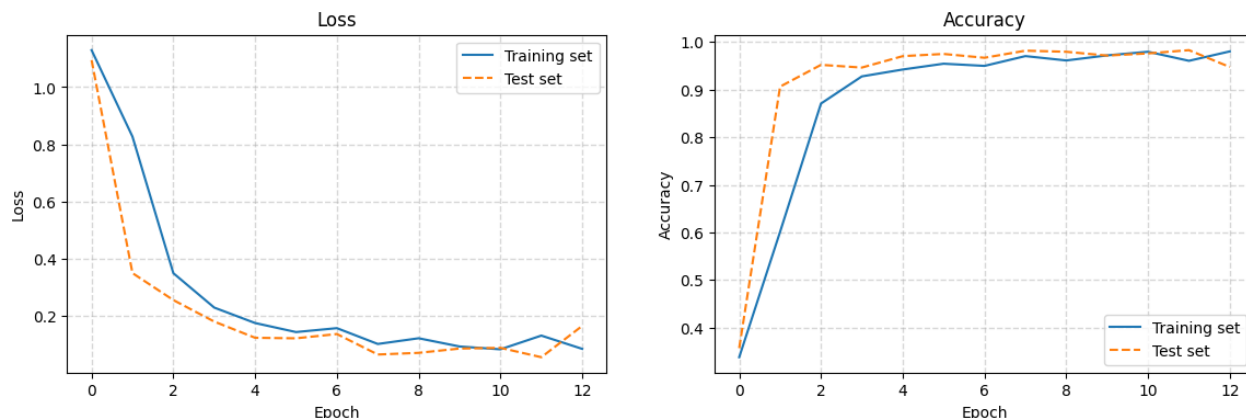
Metrik Kinerja

1. Akurasi: $(TP + TN) / (TP + TN + FP + FN) = (115 + 104 + 104) / (115 + 104 + 104 + 28 + 6) = 93.75\%$.
2. Precision (gunting): $TP / (TP + FP) = 76 / (76 + 28) = 0.7313$.
3. Recall (gunting): $TP / (TP + FN) = 76 / (76 + 104) = 0.4247$.
4. F1-Score (gunting): $2 * (Precision * Recall) / (Precision + Recall) = 2 * (0.7313 * 0.4247) / (0.7313 + 0.4247) = 0.5242$.

Berdasarkan confusion matrix dan metrik kinerja di atas, dapat disimpulkan bahwa model klasifikasi ini memiliki performa yang cukup baik dalam mengklasifikasikan objek sebagai "gunting", "batu", dan "kertas". Akurasi model mencapai 93.75%, dengan precision dan recall yang cukup tinggi untuk setiap kelas.

3.3 Pembahasan

Visualisasi sejarah pelatihan model (training history) dari model neural network yang telah dilatih. Fungsi ini menghasilkan plot yang menunjukkan perkembangan loss dan akurasi selama beberapa epoch pelatihan, baik untuk data pelatihan maupun data validasi dapat dilihat pada gambar 3.



Gambar 3. Visualisasi sejarah pelatihan model

Pada Gambar 3 ini menampilkan dua grafik: Loss vs. Epoch dan Accuracy vs. Epoch, yang umum digunakan untuk mengevaluasi kinerja model pembelajaran mesin selama pelatihan.

Loss vs. Epoch (grafik kiri)

1. Sumbu Y (Loss): Mengukur kesalahan antara nilai prediksi dan nilai aktual. Loss yang lebih rendah menunjukkan kecocokan model yang lebih baik.
2. Sumbu X (Epoch): Jumlah kali seluruh dataset pelatihan telah melewati model.
3. Garis Biru (Training set): Loss yang dihitung pada set pelatihan.
4. Garis Putus-putus Oranye (Test set): Loss yang dihitung pada set pengujian.

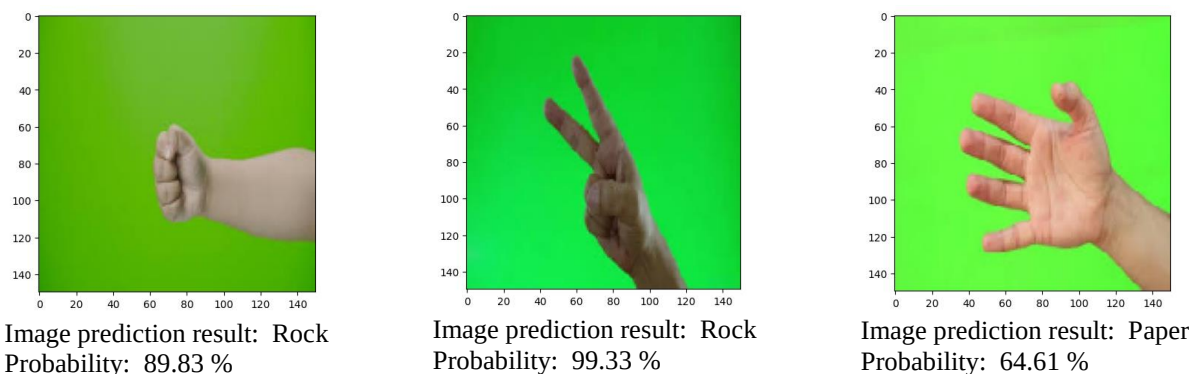
Baik loss pada pelatihan maupun pengujian menurun dengan cepat selama beberapa epoch pertama, menunjukkan bahwa model belajar dengan efektif. Setelah sekitar 5 epoch, loss stabil, menunjukkan bahwa kinerja model mulai konvergen. Pada akhir periode pelatihan, terdapat sedikit peningkatan pada loss pengujian, mengindikasikan potensi overfitting, dimana model mulai sedikit lebih buruk pada data pengujian.

Accuracy vs. Epoch (grafik kanan)

1. Sumbu Y (Accuracy): Proporsi prediksi yang benar dari total instance. Akurasi yang lebih tinggi menunjukkan kinerja yang lebih baik.
2. Sumbu X (Epoch): Jumlah kali seluruh dataset pelatihan telah melewati model.
3. Garis Biru (Training set): Akurasi pada set pelatihan.
4. Garis Putus-putus Oranye (Test set): Akurasi pada set pengujian.

Baik akurasi pelatihan maupun pengujian meningkat dengan cepat pada beberapa epoch pertama. Setelah sekitar 5 epoch, akurasi stabil, dengan model mencapai akurasi tinggi pada kedua set pelatihan dan pengujian. Terdapat sedikit penurunan pada akurasi pengujian di akhir, yang juga bisa mengindikasikan sedikit overfitting karena kinerja model pada data yang tidak terlihat tidak terus meningkat.

Model menunjukkan perilaku pembelajaran yang baik, dengan loss yang menurun dan akurasi yang meningkat pada epoch awal. Terdapat sedikit overfitting yang diamati menjelang akhir pelatihan, yang ditunjukkan oleh peningkatan loss pada pengujian dan sedikit penurunan akurasi pengujian. Kinerja keseluruhan model baik, yang ditunjukkan oleh akurasi tinggi dan loss rendah pada kedua set pelatihan dan pengujian setelah beberapa epoch. Hasil prediksi dari model CNN yang telah dilatih menunjukkan hasil yang beragam. Kemampuan model dalam memprediksi citra baru yang belum dikenal dapat dilihat pada gambar 4.



Gambar 4. Hasil prediksi model CNN

Pada Gambar 3 tersebut menunjukkan hasil prediksi model Convolutional Neural Network (CNN) untuk permainan "Batu, Gunting, Kertas". Terdapat tiga gambar tangan dengan prediksi dan probabilitas yang dihasilkan oleh model CNN. Gambar pertama (kiri) diprediksi sebagai Rock (Batu) dengan probabilitas 89.83% yaitu gambar tangan

menggenggam, sesuai dengan gerakan "batu". Pada Gambar kedua (tengah) diprediksi sebagai Rock (Batu) dengan probabilitas: 99.33% yaitu gambar tangan dengan dua jari terangkat (simbol "gunting"), namun diprediksi sebagai "batu" oleh model. Pada gambar ketiga (kanan) diprediksi sebagai Paper (Kertas) dengan probabilitas: 64.61% yaitu gambar tangan terbuka (simbol "kertas"), yang diprediksi dengan benar oleh model.

Ketidakakuratan terlihat pada gambar kedua (tengah), di mana model CNN memprediksi gerakan "gunting" sebagai "batu". Ada beberapa alasan mengapa prediksi ini tidak akurat:

1. **Training Data yang Tidak Seimbang:**
Jika data pelatihan lebih banyak berisi gambar "batu" dibandingkan "gunting" atau "kertas", model mungkin lebih condong untuk memprediksi "batu" secara lebih sering.
2. **Variasi dalam Pose Tangan:**
Variasi kecil dalam pose tangan atau sudut pengambilan gambar dapat menyebabkan model kesulitan dalam mengenali bentuk tertentu. Misalnya, model mungkin kesulitan membedakan antara dua jari yang terangkat (gunting) dan jari-jari yang tergenggam sebagian (batu).
3. **Preprocessing Data yang Tidak Optimal:**
Proses preprocessing seperti normalisasi, cropping, atau augmentasi mungkin tidak dilakukan dengan cukup baik, sehingga model mengalami kesulitan dalam mengenali pola pada gambar.
4. **Overfitting pada Dataset Latihan:**
Model mungkin telah overfit pada dataset pelatihan sehingga kurang mampu menggeneralisasi pada data baru. Ini sering terjadi jika model terlalu kompleks atau data latihannya kurang beragam.
5. **Arsitektur Model yang Kurang Tepat:**
Arsitektur CNN yang digunakan mungkin tidak optimal untuk tugas klasifikasi ini. Misalnya, jumlah layer, filter, atau ukuran kernel yang tidak sesuai bisa mempengaruhi kemampuan model dalam mengekstrak fitur dari gambar dengan baik.

Untuk meningkatkan akurasi prediksi, beberapa langkah yang dapat dilakukan antara lain:

1. Mengumpulkan lebih banyak data latih yang seimbang untuk ketiga kelas (batu, gunting, kertas).
2. Melakukan augmentasi data untuk meningkatkan variasi dalam dataset.
3. Menyesuaikan arsitektur model CNN agar lebih mampu menangkap fitur-fitur yang relevan.
4. Menggunakan teknik regularisasi seperti dropout untuk mengurangi overfitting.
5. Evaluasi dan tuning preprocessing steps, termasuk normalisasi dan cropping gambar, agar fitur penting lebih mudah diekstraksi oleh model.

Dengan langkah-langkah tersebut, diharapkan model CNN bisa lebih akurat dalam melakukan prediksi pada gambar baru.

4. KESIMPULAN

Hasil eksperimen menunjukkan bahwa kombinasi teknik optimasi ini secara signifikan meningkatkan performa model CNN pada beberapa dataset citra standar. Temuan ini menegaskan pentingnya penggunaan strategi optimasi yang holistik dalam pengembangan model CNN untuk aplikasi klasifikasi dan prediksi citra. Namun pada beberapa percobaan model yang telah dilatih masih terdapat kesalahan prediksi. Hal ini dapat disebabkan oleh beberapa faktor seperti overfitting atau underfitting, kualitas dan kuantitas data, keragaman data, arsitektur model, dan metode optimasi. Untuk itu masih diperlukan optimasi lainnya baik pada persiapan data, penentuan metode optimasi, serta pembersihan data.

REFERENCES

- [1] B. Mahesh, "Machine Learning Algorithms-A Review," Int. J. Sci. Res., 2018, doi: 10.21275/ART20203995.
- [2] M. Arsal, B. Agus Wardijono, and D. Anggraini, "Jurnal Nasional Teknologi dan Sistem Informasi Face Recognition Untuk Akses Pegawai Bank Menggunakan Deep Learning Dengan Metode CNN", doi: 10.25077/TEKNOSI.v6i1.2020.55-63.
- [3] E. Rasywir, R. Sinaga, Y. Pratama, U. Dinamika, and B. Jambi, "Analisis dan Implementasi Diagnosis Penyakit Sawit dengan Metode Convolutional Neural Network (CNN)," vol. 22, no. 2, 2020, doi: 10.31294/p.v2i12.
- [4] A. Peryanto, A. Yudhana, and D. R. Umar, "Rancang Bangun Klasifikasi Citra Dengan Teknologi Deep Learning Berbasis Metode Convolutional Neural Network," Jurnal, vol. 8, pp. 2089-5615, 2019, Accessed: Jun. 01, 2024. [Online]. Available: <https://www.mathworks.com/discovery/convolutional-neural-network.html>
- [5] P. : Randi et al., "ALGORITMA PEMBELAJARAN MESIN (Dasar, Teknik, dan Aplikasi)", Accessed: May 23, 2024. [Online]. Available: www.buku.sonpedia.com
- [6] T. Shafira, "Implementasi Convolutional Neural Networks untuk Klasifikasi Citra Tomat Menggunakan Keras," Mar. 2018, Accessed: May 23, 2024. [Online]. Available: <https://dspace.uui.ac.id/handle/123456789/6345>
- [7] I. N. U. R. ALAM, "METODE TRANSFER LEARNING PADA DEEP CONVOLUTIONAL NEURAL NETWORK (DCNN) UNTUK PENGENALAN EKSPRESI WAJAH".
- [8] C. B. LIMBOING, "IMPLEMENTASI DEEP LEARNING UNTUK MENGANALISA BERBAGAI MACAM DAUN," Sep. 2022.
- [9] S. Dolnicar et al., "PENGENALAN DAN KLASIFIKASI RAGAM KUE INDONESIA MENGGUNAKAN ARSITEKTUR RESNET50V2 PADA CONVOLUTIONAL NEURAL NETWORK (CNN)," Why We Need the Journal of Interactive Advertising, vol. 3, no. 1, p. 45, 1997. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0160738315000444%0Ahttp://www.sciencedirect.com/science/article/pii/S0>

- 160738315000444%250Ahttp://eprints.lancs.ac.uk/48376/%255Cnhttp://dx.doi.org/10.1002/zamm.19630430112%250Ahttp://www.sciencedirect.com/
- [10] Julien de la Bruère-Terreault, “Rock-Paper-Scissors Images.” <https://www.kaggle.com/datasets/drgfreeman/rockpaperscissors> (accessed May 23, 2024).
- [11] C. Shorten and T. M. Khoshgoftaar, “A survey on Image Data Augmentation for Deep Learning,” *J. Big Data*, vol. 6, no. 1, pp. 1–48, Dec. 2019, doi: 10.1186/S40537-019-0197-0/FIGURES/33.
- [12] J. Ghosh and S. Gupta, “ADAM Optimizer and CATEGORICAL CROSSENTROPY Loss Function-Based CNN Method for Diagnosing Colorectal Cancer,” *Proc. Int. Conf. Comput. Intell. Sustain. Eng. Solut. CISES 2023*, pp. 470–474, 2023, doi: 10.1109/CISES58720.2023.10183491.
- [13] U. N. Oktaviana, R. Hendrawan, A. D. K. Annas, and G. W. Wicaksono, “Klasifikasi Penyakit Padi berdasarkan Citra Daun Menggunakan Model Terlatih Resnet101,” *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 6, pp. 1216–1222, Dec. 2021, doi: 10.29207/RESTI.V5I6.3607.
- [14] S. A. Wulandari, M. Ma’ruf, A. R. Priyatno, N. Halimun, Z. M. Abdulah, and U. Amartiwi, “DjunkGo: A Mobile Application for Trash Classification with VGG16 Algorithm,” *GMPI Conf. Ser.*, vol. 2, pp. 67–72, Jan. 2023, doi: 10.53889/GMPICS.V2.175.
- [15] S. Poudel and P. Poudyal, “Classification of Waste Materials using CNN Based on Transfer Learning,” *ACM Int. Conf. Proceeding Ser.*, pp. 29–33, Dec. 2022, doi: 10.1145/3574318.3574345.
- [16] A. Darugutni and H. Marcos, “KLASIFIKASI PERMAINAN BATU KERTAS GUNTING MENGGUNAKAN ALGORITMA CONVOLUTION NEURAL NETWORK (CNN),” *Method. J. Tek. Inform. dan Sist. Inf.*, vol. 9, no. 1, pp. 1–3, Mar. 2023, doi: 10.47065/bits.v3i3.1143.
- [17] Fitriani, “KLASIFIKASI JENIS BUNGA DENGAN MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORK (CNN),” *TEKNIMEDIA*, vol. 2, no. 2, pp. 64–68, 2021.
- [18] M. Malik et al., “Waste Classification for Sustainable Development Using Image Recognition with Deep Learning Neural Network Models,” *Sustain.*, vol. 14, no. 12, Jun. 2022, doi: 10.3390/SU14127222.
- [19] A. Ibnul Rasidi, Y. A. H. Pasaribu, A. Ziqri, and F. D. Adhinata, “Klasifikasi Sampah Organik dan Non-Organik Menggunakan Convolutional Neural Network,” *J. Tek. Inform. dan Sist. Inf.*, vol. 8, no. 1, Apr. 2022, doi: 10.28932/JUTISI.V8I1.4314.