

Identifikasi Ekspresi Wajah Manusia Menggunakan Algoritma Grey Wolf Optimizer dan Convolutional Neural Network

Ni'matur Rohim¹, Andi Sunyoto², Kusnawi³

^{1,2,3}Universitas Amikom Yogyakarta, Jl. Ring Road Utara, Ngringin, Condongcatur, Daerah Istimewa Yogyakarta 55281, Indonesia

Article Info

Article history:

Diterima 21 Februari 2024

Revisi 26 Februari 2024

Diterbitkan 5 April, 2024

Keywords:

Convolutional Neural Network
Identifikasi Ekspresi Wajah
Grey Wolf Optimizer

ABSTRAK

Penelitian ini mengeksplorasi penggunaan algoritma Grey Wolf Optimizer (GWO) untuk mengoptimalkan parameter pada Convolutional Neural Network (CNN) dalam mengenali ekspresi wajah manusia. Ekspresi wajah adalah aspek penting dalam komunikasi manusia, dan pengenalan ekspresi tersebut menjadi semakin vital dalam interaksi manusia-mesin dan bidang kesehatan psikologi. Metode deep learning, terutama CNN, telah terbukti efektif dalam mengklasifikasikan ekspresi manusia, meskipun masih menghadapi beberapa tantangan, seperti pengaturan parameter yang rumit dan kebutuhan akan data yang besar. Penelitian ini bertujuan untuk mencari parameter optimal untuk meningkatkan kinerja CNN dalam mengenali ekspresi wajah menggunakan algoritma GWO. Data yang digunakan adalah dataset *Facial Expression Recognition 2013 (FER-2013)*, dengan total 600 citra wajah yang dibagi menjadi tiga kelas: *happy*, *sad*, dan *angry*. Pendekatan yang diusulkan mencakup *preprocessing* data, pencarian parameter arsitektur CNN menggunakan GWO, pembuatan model CNN, dan pengujian model menggunakan data *testing*. Hasil pengujian menunjukkan bahwa dengan parameter optimal, model CNN mencapai akurasi yang baik, dengan nilai akurasi 79% pada data *training*, 60% pada data *validation*, dan rata-rata akurasi 77% pada data *testing*. Penelitian ini menyoroti pentingnya penanganan yang cermat dalam menentukan parameter untuk memastikan hasil yang optimal dalam pengenalan ekspresi wajah manusia menggunakan CNN.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Ni'matur Rohim,

Universitas Amikom Yogyakarta

ninik@students.amikom.ac.id

1. PENDAHULUAN

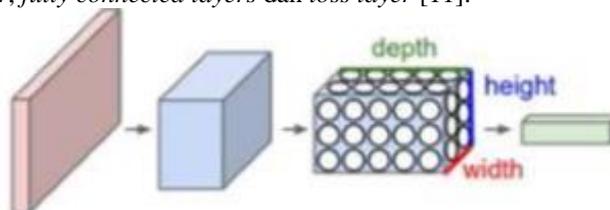
Wajah merupakan bagian tubuh yang dapat mengindikasikan emosi seseorang melalui ekspresinya, apakah ia bahagia (*happy*), sedih (*sad*), marah (*angry*), jijik (*discusting*), terkejut (*sureprise*) ataupun biasa saja (*neutral*) [1]. Ekspresi manusia merupakan salah satu cara yang paling penting dalam komunikasi manusia. Dalam berbagai situasi, ekspresi wajah dapat membantu mengirimkan pesan yang penting bagi pemahaman orang lain, misalnya untuk menunjukkan emosi, sikap, dan niat [2]. Pengenalan ekspresi wajah menjadi penting seiring dengan meningkatnya teknologi yang ada pada komputer, telepon genggam, robot dan sebagainya. Peningkatan tersebut membuat interaksi antara manusia dan mesin semakin tidak terelakkan. Sebagai contoh sistem belanja online saat ini, dengan menggunakan ekspresi wajah, maka dapat dibuat suatu sistem yang mampu mengenali kepuasan pelanggan salah satunya. Selain itu, pada bidang kesehatan psikologi, interaksi manusia dan komputer (robot) dan sebagainya [1], [3]. Karena pentingnya ekspresi wajah dalam komunikasi manusia, studi tentang pengenalan ekspresi wajah telah menjadi topik yang penting dalam bidang pengolahan citra dan pengenalan pola. Dalam beberapa tahun terakhir, *deep learning* telah menjadi teknik yang sangat populer dalam mengklasifikasikan ekspresi manusia. Salah satu jenis *deep learning* yang sering digunakan untuk pengenalan ekspresi manusia adalah *Convolutional Neural Network (CNN)*.

CNN merupakan salah satu jenis *neural network* yang khusus dirancang untuk memproses data dengan struktur grid seperti gambar. CNN memungkinkan sebuah sistem untuk mengenali pola pada gambar dengan cara yang lebih efektif dan akurat dibandingkan dengan teknik pengenalan pola tradisional [2], [4]. CNN telah berhasil digunakan dalam berbagai aplikasi seperti klasifikasi gambar, pengenalan wajah, dan segmentasi gambar. Dalam konteks pengenalan ekspresi manusia, CNN digunakan untuk memproses gambar wajah manusia dan mengklasifikasikan ekspresi wajah yang ditampilkan. CNN akan memproses gambar wajah melalui beberapa layer untuk menghasilkan fitur - fitur yang dapat digunakan untuk mengklasifikasikan ekspresi wajah. Setelah fitur - fitur dihasilkan, CNN akan menghubungkan layer-layer tersebut untuk menghasilkan output yang merepresentasikan klasifikasi ekspresi wajah.

Penerapan *deep learning* menggunakan CNN pada pengenalan ekspresi wajah manusia telah menunjukkan hasil yang sangat baik [2], [5], [6]. Beberapa penelitian telah menunjukkan akurasi klasifikasi yang bagus dibandingkan dengan teknik pengenalan pola tradisional, seperti yang telah dilakukan oleh [6] dalam penelitiannya tentang klasifikasi ekspresi wajah menggunakan algoritma CNN dengan menggunakan beberapa parameter *optimizer* yang hasilnya sangat baik dalam proses *training* data namun memiliki kekurangan dalam proses pengujian data *testing*. Selain itu, CNN juga memiliki kelebihan dalam memproses gambar dengan variasi pose, pencahayaan, dan kualitas gambar yang berbeda. Namun, meskipun CNN menunjukkan hasil yang sangat baik dalam pengenalan ekspresi manusia, terdapat beberapa tantangan yang harus diatasi. Salah satu tantangan utama adalah kebutuhan akan data yang besar dan berkualitas untuk melatih CNN. Selain itu, CNN juga memiliki banyak parameter yang harus diatur dengan hati-hati untuk mendapatkan hasil yang baik. Sejauh ini pemilihan parameter yang optimal untuk algoritma CNN seperti parameter *loss function*, *number of convolution filters*, *optimizer function* dan parameter lainnya dilakukan secara manual dengan cara *trial and error* sehingga membutuhkan waktu yang lama untuk memperoleh parameter yang paling optimal [7]. Oleh karena itu, identifikasi ekspresi wajah manusia menggunakan metode *Convolutional Neural Network* membutuhkan penanganan yang cermat dan teliti untuk memperoleh hasil akurasi yang maksimal.

Penelitian ini mencoba untuk mencari parameter yang lebih optimal untuk metode CNN serta untuk menguji kinerja algoritma *Grey Wolf Optimizer* (GWO) dalam mencari parameter yang optimal untuk algoritma CNN pada klasifikasi Ekspresi Wajah Manusia. Algoritma GWO merupakan algoritma *metaheuristik* yang tergolong baru serta mampu memberikan hasil yang kompetitif, dimana eksplorasi pencarian spesiesnya lebih luas serta dapat menghindari terjebaknya lokal optimum [7], [8]. Algoritma GWO juga telah terbukti memberikan hasil yang bagus untuk otomasi pemilihan atribut yang optimal dalam proses klasifikasi gambar dengan menggunakan algoritma CNN [9].

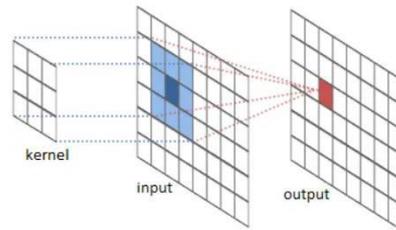
Salah satu komponen dari *deep learning* yang banyak digunakan dalam *computer vision* adalah metode yang disebut ConvNet atau CNN. Mirip dengan jaringan saraf tiruan / *artificial neural network* (ANN) konvensional, CNN memiliki struktur yang terdiri dari neuron yang terhubung dalam suatu *layer*. Namun, berbeda dengan ANN, CNN menggunakan proses konvolusi untuk mengekstraksi fitur dari gambar [10]. Penggunaan luas CNN dalam *computer vision* tercermin dari berbagai arsitektur yang dikembangkan untuk menangani berbagai masalah. Beberapa arsitektur populer meliputi AlexNet, VGG, Google Inception, dan lain-lain. Arsitektur CNN sendiri terdiri dari *hidden layers* yang mencakup *convolutional layers*, *pooling layers*, *normalization*, *ReLU layer*, *fully connected layers* dan *loss layer* [11].



Gambar 1. Arsitektur CNN secara umum

a. Convolutional Layer

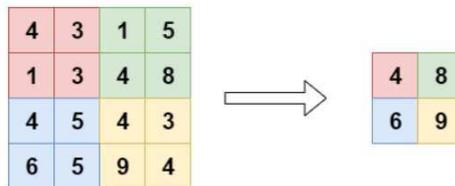
Convolutional Layer adalah komponen kunci dari CNN, karena sebagian besar perhitungan pada CNN terjadi di lapisan ini. Operasinya mirip dengan konvolusi yang umumnya digunakan dalam pemrosesan citra, di mana terdapat kernel dan sub-citra. Kernel yang digunakan dalam CNN biasanya memiliki dimensi 3x3. Selanjutnya, operasi konvolusi dilakukan untuk setiap sub-citra yang memiliki ukuran yang sama dengan kernel.



Gambar 2. Convolutional Layer

b. Pooling Layer

Pooling Layer adalah elemen yang dimaksudkan untuk mengurangi dimensi citra (pengurangan resolusi) dengan tujuan mempermudah konvolusi pada lapisan berikutnya. Langkah ini dapat dilakukan melalui berbagai teknik, termasuk *max pooling* (mengambil nilai maksimum dari sub-citra), *average pooling* (mengambil nilai rata-rata dari sub-citra), dan sebagainya.



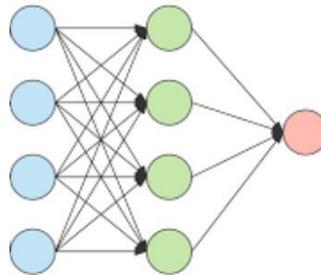
Gambar 3. Pooling Layer (max pooling)

c. ReLu Layer

Lapisan *Rectified Linear Unit* (ReLU) berfungsi untuk meningkatkan non-linearitas dari fungsi keputusan.

d. Fully Connected Layer

Di lapisan ini, setiap *neuron* terhubung sepenuhnya (*fully connected*) seperti yang ditemukan dalam *multi-layer perceptron* (MLP). Di bagian ini, terdapat parameter bobot dan bias yang digunakan untuk menyesuaikan data, mirip dengan yang dilakukan oleh MLP.



Gambar 4. Fully Connected Layer

e. Loss Layer

Ini merupakan lapisan terakhir dalam CNN yang berperan untuk menilai ketidakcocokan antara hasil prediksi dan label target selama proses pelatihan, yang mempengaruhi penyesuaian bobot dan bias. Terdapat berbagai fungsi yang bisa diterapkan di lapisan ini, seperti *sigmoid cross-entropy loss*, *softmax loss*, *euclidean loss*, dan lainnya.

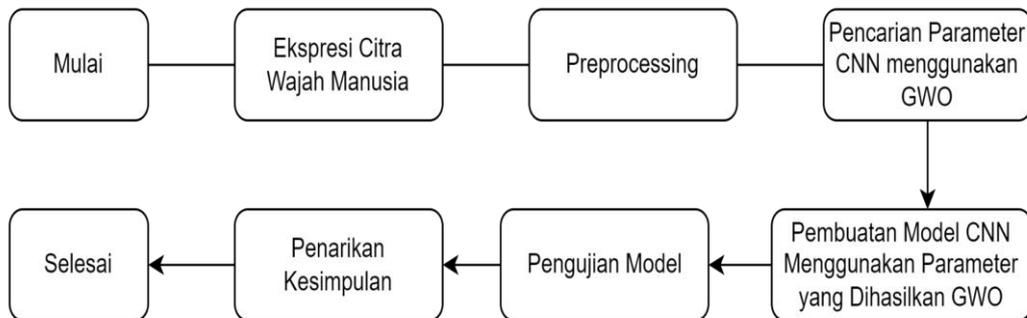
GWO merupakan algoritma optimasi metaheuristik yang terinspirasi dari gerakan dan hierarki sosial dalam kelompok serigala. Algoritma ini pertama kali diperkenalkan oleh Mirjalili et al. pada tahun 2014 [7], [8], [9] GWO bekerja dengan mengoptimalkan suatu fungsi tujuan dengan menggerakkan serigala dalam kelompoknya. Adapun tahapan-tahapan pada algoritma ini yakni:

1. Inisialisasi: Menciptakan sebuah populasi wolf secara acak dalam batasan solusi yang telah ditentukan.
2. Evaluasi: Menghitung nilai fungsi tujuan pada setiap wolf dalam populasi.
3. Pemilihan pemimpin: Memilih wolf dengan nilai fungsi tujuan terbaik (alpha), wolf dengan nilai kedua terbaik (beta), dan wolf dengan nilai ketiga terbaik (delta).
4. Pembaruan posisi wolf: Setiap wolf mengikuti gerakan dan posisi pemimpinnya dalam kelompoknya.

5. Pembaruan batasan solusi: Mengecek setiap wolf apakah masih berada dalam batasan solusi, jika tidak maka posisi wolf diperbaiki sesuai batasan solusi.
6. Evaluasi: Menghitung nilai fungsi tujuan pada setiap wolf setelah pembaruan posisi dan batasan solusi.
7. Kondisi berhenti: Menghentikan iterasi ketika tercapai kondisi berhenti yang telah ditentukan atau mencapai batas maksimum iterasi.

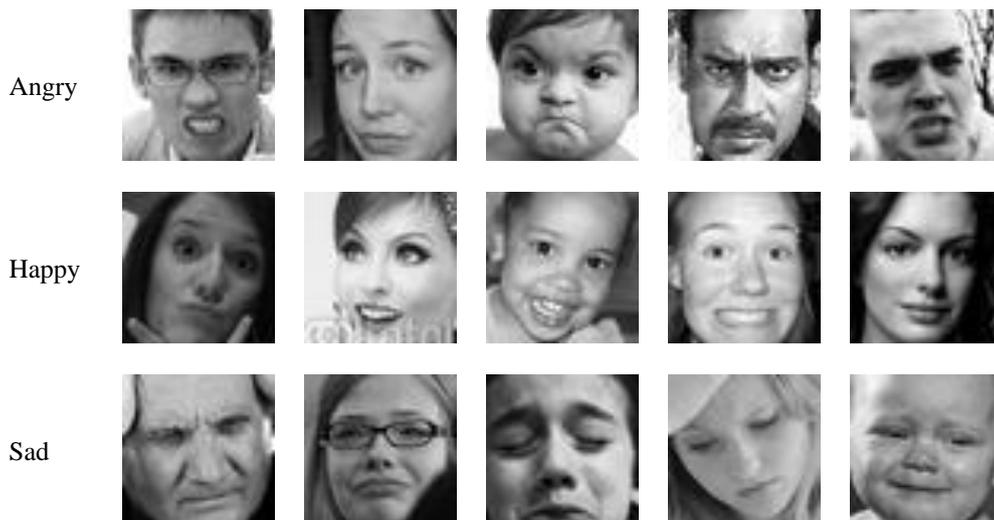
2. METODE

Pendekatan yang digunakan pada penelitian ini menggunakan pendekatan kuantitatif sesuai dengan alur penelitian yang telah dibuat untuk memperoleh hasil dari proses klasifikasi Ekspresi Wajah Manusia yang meliputi ekspresi *happy*, *sad* dan *angry*. Pada penelitian ini mengusulkan pendekatan algoritma CNN dengan optimasi algoritma GWO untuk memperoleh parameter terbaik yang akan digunakan pada metode CNN untuk identifikasi citra gambar Ekspresi Wajah Manusia. Setelah memperoleh parameter yang optimal untuk algoritma CNN selanjutnya yakni melakukan pengujian dataset terhadap model yang telah dibuat menggunakan parameter optimal yang dihasilkan dari algoritma GWO. Pada tahapan ini berisi langkah-langkah yang benar agar penelitian ini dapat berjalan dengan efektif. Berikut alur penelitian ini dapat dilihat pada gambar bawah ini.



Gambar 1. Bagan Alur Penelitian

Dataset citra wajah yang digunakan dalam penelitian ini yakni dataset Facial Expression Recognition 2013(FER-2013). Jumlah dataset yang digunakan dalam penelitian ini menggunakan data sebanyak 600 citra wajah, untuk masing-masing kelas (*happy*, *sad*, dan *angry*) terdiri dari 200 citra serta citra wajah berukuran 48x48 dengan citra *greyscale*. Data yang telah terkumpul dibagi menjadi 3 bagian untuk setiap kelas yakni data *training*, *validation*, dan *testing* dengan komposisi data *training* (480), *validation* (45), dan data *testing* (75). Adapun beberapa contoh dataset yang digunakan yakni sebagai berikut:



Gambar 2. Sample Dataset Citra Ekspresi Wajah

Data gambar ekspresi wajah manusia akan diproses sebelum digunakan dalam pelatihan dan pengujian model, *preprocessing* ini meliputi langkah - langkah normalisasi intensitas piksel, dan *resizing* (penyesuaian ukuran gambar). Langkah - langkah ini bertujuan untuk mempersiapkan data sehingga sesuai kebutuhan model CNN. Setelah dilakukan *preprocessing* maka proses selanjutnya adalah melakukan pencarian parameter arsitektur CNN menggunakan GWO. Parameter CNN yang dicari meliputi a) nilai dari 3 layer yang digunakan (layer 1, layer 2, dan layer 3), b) nilai ukuran filter, c) nilai *dense layer*, d) *learning rate*, dan e) *epoch*. Adapun tahapan dari pencarian parameter CNN menggunakan GWO yakni :

1. Persiapan data training dan validation untuk pembuatan model CNN
2. Menentukan arsitektur CNN yang akan digunakan, termasuk jumlah lapisan konvolusi, ukuran kernel, jumlah neuron di setiap lapisan tersembunyi, dan fungsi aktivasi yang akan digunakan.
3. Menentukan parameter-parameter yang ingin dioptimalkan, dalam penelitian ini, parameter yang akan dioptimalkan yakni meliputi a) nilai dari 3 layer yang digunakan (layer 1, layer 2, dan layer 3), b) nilai ukuran filter, c) nilai dense layer, d) learning rate, dan e) epoch.

Segmen Kode 1. Fungsi Parameter Arsitektur CNN yang dicari

```

1. def fitness_function(position, callback):
2.     learning_rate,n1,n2,n3,n4,kernel_size,epochs,_,_,_ = position
3.     n1,n2,n3,n4,kernel_size,epochs=int(n1),int(n2),int(n3),int(n4),int(kernel_size),int(epochs)
4.     print(learning_rate,n1,n2,n3,n4,kernel_size,epochs)
5.     model = tf.keras.models.Sequential([
6.         tf.keras.layers.Conv2D(n1, (kernel_size,kernel_size), activation='relu',
7.         input_shape=(48, 48, 3), kernel_regularizer=tf.keras.regularizers.l2(0.001)),
8.         tf.keras.layers.MaxPooling2D(2, 2),
9.         tf.keras.layers.BatchNormalization(),
10.        tf.keras.layers.Conv2D(n2, (kernel_size,kernel_size), activation='relu'),
11.        tf.keras.layers.MaxPooling2D(2,2),
12.        tf.keras.layers.BatchNormalization(),
13.        tf.keras.layers.Conv2D(n3, (kernel_size,kernel_size), activation='relu'),
14.        tf.keras.layers.MaxPooling2D(2,2),
15.        tf.keras.layers.BatchNormalization(),
16.        tf.keras.layers.Flatten(),
17.        tf.keras.layers.Dense(n4, activation='relu'),
18.        tf.keras.layers.BatchNormalization(),
19.        tf.keras.layers.Dense(3, activation='softmax')])
20.    optimizer = tf.optimizers.Adam(learning_rate=learning_rate)
21.    model.compile(loss='sparse_categorical_crossentropy',
22.    optimizer=optimizer, metrics=['accuracy'])
23.    model.fit(x_train,y_train, epochs=int(epochs), verbose=0)
24.    _, accuracy = model.evaluate(x_test, y_test, verbose=0)
25.    callback.update(model, accuracy)
26.    return accuracy
27.
28. def objective_function(solution):
29.     accuracy = fitness_function(solution, my_callback)
30.     return -accuracy

```

4. Mengimplementasikan Algoritma GWO untuk pencarian parameter yang telah ditentukan pada segmen kode program diatas dengan memanggil fungsi `objective_function` serta penentuan nilai minimum dan maksimum parameter yang akan dicari. Selain itu pada tahapan ini pula menentukan epoch yang akan dicari pula, seperti terlihat pada segmen kode berikut:

Segmen Kode 2. Implementasi Algoritma GWO

```

1. problem_dict1 = {
2.     "fit_func": objective_function,
3.     'lb':[0.001,16,32,64,128,3,5,1,1,1],
4.     'ub':[0.1,32,64,128,512,5,30,1,1,1],
5.     "minmax": "min"}
6.
7.
8. epoch = 100
9. pop_size = 10
10. gwo = OriginalGWO(epoch, pop_size)
11. best_position, best_fitness = gwo.solve(problem_dict1)
12. print(f"Solution: {best_position}, Fitness: {best_fitness}")

```

5. Setelah proses pencarian parameter menggunakan Algoritma GWO selesai. Maka dapat diketahui parameter yang optimal yang dapat digunakan untuk algoritma CNN dalam Klasifikasi Citra Ekspresi Wajah. Adapun parameter yang dihasilkan yakni:
 - Layer1 - 23 filters
 - Layer2 - 48 filters
 - Layer3 - 97 filters
 - Dense Layer- 236 neurons
 - All layers have a filter size of (4,4)
 - learning_rate=0.001430901114084654
 - epochs=20

Adapun hasil arsitektur CNN menggunakan parameter diatas yakni sebagai berikut:

| Layer (type) | Output Shape | Param # |
|---|--------------------|---------|
| conv2d (Conv2D) | (None, 46, 46, 23) | 644 |
| batch_normalization (Batch Normalization) | (None, 46, 46, 23) | 92 |
| max_pooling2d (MaxPooling2D) | (None, 23, 23, 23) | 0 |
| conv2d_1 (Conv2D) | (None, 21, 21, 48) | 9984 |
| batch_normalization_1 (Batch Normalization) | (None, 21, 21, 48) | 192 |
| max_pooling2d_1 (MaxPooling2D) | (None, 10, 10, 48) | 0 |
| conv2d_2 (Conv2D) | (None, 7, 7, 97) | 74593 |
| batch_normalization_2 (Batch Normalization) | (None, 7, 7, 97) | 388 |
| max_pooling2d_2 (MaxPooling2D) | (None, 3, 3, 97) | 0 |
| flatten (Flatten) | (None, 873) | 0 |
| dense (Dense) | (None, 236) | 206264 |
| dense_1 (Dense) | (None, 3) | 711 |

Total params: 292868 (1.12 MB)
 Trainable params: 292532 (1.12 MB)
 Non-trainable params: 336 (1.31 KB)

Gambar 2. Arsitektur Model CNN Menggunakan Parameter Hasil GWO

6. Tahapan terakhir yakni menggunakan parameter yang telah diperoleh dari hasil GWO pada arsitektur CNN dan melakukan evaluasi model / uji coba terhadap data *testing*.

Model CNN yang akan dibuat dalam penelitian memiliki 3 lapisan konvolusi dengan fitur berukuran 4x4, menerapkan fungsi aktivasi ReLU, serta melakukan *max pooling* menggunakan kernel 2x2 serta menggunakan batch normalization di setiap layer-nya. Setelah itu, akan dilakukan proses *flatten*. Terdapat juga 2 lapisan dense dengan fungsi aktivasi ReLU dan sigmoid, diikuti dengan output yang memiliki 3 kelas menggunakan fungsi aktivasi sigmoid. Dataset yang digunakan dalam pembuatan model menggunakan data *training* dan data *validation* sesuai pembagian diatas. Setelah pembuatan model dilakukan tahapan yang dilakukan selanjutnya yakni pengujian model menggunakan data *testing* yang telah disiapkan sebelumnya. Setelah proses pengujian model menggunakan data *testing*, tahapan yang terakhir adalah penarikan kesimpulan dan analisa hasil proses pengujian yang telah dilakukan dengan menggunakan *confusion matrix*. Tabel *confusion matrix* dapat dilihat seperti berikut:

Tabel 1. *Confusion Matrix*

| Confusion Matrix | Prediksi | | |
|------------------|-------------|-------------|-------------|
| | Angry | Happy | Sad |
| Aktual Angry | True Angry | False Angry | False Angry |
| Aktual Happy | False Happy | True Happy | False Happy |
| Aktual Sad | False Sad | False Sad | True Sad |

3. HASIL DAN PEMBAHASAN

Eksresi wajah yang direpresentasikan dalam bentuk citra dengan ukuran 48 x 48 piksel dan akan mengalami konvolusi pada lapisan konvolusi menggunakan filter berukuran 4x4 dan fungsi aktivasi ReLU. Langkah ini dijalankan dengan maksud untuk menghasilkan 23 fitur yang merepresentasikan ciri-ciri penting dari ekspresi wajah tersebut. Selanjutnya, proses *pooling* dilakukan dengan menerapkan metode *max pooling* berukuran 2x2, yang bertujuan untuk mengurangi dimensi citra dengan melakukan operasi *down-sampling*.

Proses berikutnya melibatkan 2 tahap konvolusi kembali, konvolusi kedua menggunakan kernel berukuran 3x3 dan fungsi aktivasi ReLU yang menghasilkan 48 fitur baru dan konvolusi ketiga menggunakan kernel berukuran 4x4 dan fungsi aktivasi ReLU yang menghasilkan 97 fitur baru, dimana setiap selesai proses konvolusi dari 2 konvolusi tersebut dilakukan proses *max pooling* dengan ukuran filter 2x2.

Tahap berikutnya merupakan melakukan proses penataan ulang sehingga fitur yang diekstraksi dapat diubah menjadi sebuah vektor berukuran 873, untuk kemudian dijadikan input bagi lapisan terhubung sepenuhnya (*fully connected layer*). Kemudian, vektor tersebut diubah menjadi 236 fitur, yang pada akhirnya akan menghasilkan output dalam bentuk tiga kelas menggunakan fungsi aktivasi sigmoid.

Pembuatan model CNN pada proses pelatihan dilakukan sebanyak 20 *epoch*. Model yang dihasilkan memperoleh nilai akurasi 79% pada data *training* dan 60% nilai akurasi pada data *validation*. Selanjutnya pengujian dilaksanakan dengan menggunakan data uji untuk setiap kelas dari model CNN yang telah dibangun dengan optimasi yang telah ditetapkan yakni optimasi Adam serta menggunakan parameter yang telah dihasilkan dari proses GWO. Hasil pengujian untuk data *testing* terhadap model yang telah dibuat dapat dilihat pada tabel berikut:

Tabel 2. Confusion Matrix Optimizer Adam

| Confusion Matrix | | Prediksi | | |
|------------------|-------|----------|-------|-----|
| | | Angry | Happy | Sad |
| Aktual | Angry | 19 | 3 | 3 |
| | Happy | 2 | 20 | 3 |
| | Sad | 3 | 5 | 17 |

Tabel 3. Hasil Akurasi, Precision, dan Recall

| Kelas | Precision | Recall | Akurasi |
|-------|-----------|--------|---------|
| Angry | 79% | 76% | |
| Happy | 71% | 80% | 77% |
| Sad | 68% | 74% | |

Berdasarkan tabel 3 diatas, rata-rata hasil akurasi dari data *testing* yang digunakan yakni 77% dengan rincian citra ekspresi wajah *angry* terklasifikasikan benar dengan nilai akurasi 76%, *happy* terklasifikasikan benar dengan nilai akurasi 80%, dan *sad* terklasifikasikan benar dengan nilai akurasi 74%. Sehingga rata-rata hasil akurasi data *testing* yakni 77%.

4. KESIMPULAN

Eksresi wajah merupakan komponen penting dalam komunikasi manusia, memungkinkan individu untuk menyampaikan emosi, sikap, dan niat mereka kepada orang lain. Dengan meningkatnya teknologi, pengenalan ekspresi wajah menjadi semakin penting, terutama dalam interaksi antara manusia dan mesin, seperti dalam sistem belanja online atau dalam bidang kesehatan psikologi. Pengenalan ekspresi wajah telah menjadi topik penelitian yang signifikan dalam pengolahan citra dan pengenalan pola, khususnya dengan penggunaan teknik *deep learning* seperti CNN. CNN telah terbukti efektif dalam mengklasifikasikan ekspresi manusia dengan akurasi yang baik, namun masih memiliki beberapa tantangan seperti kebutuhan akan data yang besar dan parameter yang harus diatur dengan hati-hati. Untuk mengatasi tantangan ini, penelitian ini mencoba mencari parameter yang optimal menggunakan algoritma GWO untuk meningkatkan kinerja CNN dalam mengenali ekspresi wajah manusia. Hasil dari penelitian ini menunjukkan bahwa dengan parameter yang optimal, model CNN dapat mencapai akurasi yang baik dalam mengklasifikasikan ekspresi wajah manusia. Namun demikian, masih diperlukan penanganan yang cermat dan teliti dalam menentukan parameter untuk memastikan hasil yang maksimal. Model yang dihasilkan memperoleh nilai akurasi 79% pada data *training* dan 60% nilai akurasi pada data *validation*, sedangkan rata-rata nilai akurasi data *testing* yakni 77%.

UCAPAN TERIMA KASIH

Terima kasih kepada para dosen pembimbing yang telah memberikan arahan dan bimbingan yang sangat berharga serta keluarga dan teman-teman yang selalu memberikan dukungan dan motivasi selama proses penulisan jurnal ini, tanpa dukungan dan bantuan dari semua pihak, saya tidak akan bisa menyelesaikan jurnal ini dengan baik.

REFERENSI

- [1] S. S. Shafira, N. Ulfa, H. A. Wibawa, and Rismiyati, "Facial Expression Recognition Using Extreme Learning Machine," in *2019 3rd International Conference on Informatics and Computational Sciences (ICICoS)*, 2019, pp. 1–6. doi: 10.1109/ICICoS48119.2019.8982443.
- [2] P. A. Nugroho, I. Fenriana, and R. Arijanto, "Implementasi Deep Learning Menggunakan Convolutional Neural Network (CNN) Pada Ekspresi Manusia," *Jurnal Algor*, vol. 2, no. 1, 2020, [Online]. Available: <https://jurnal.buddhidharma.ac.id/index.php/algor/index>
- [3] R. Julianto and D. Alamsyah, "Pengenalan Ekspresi Wajah Menggunakan Metode SVM Dengan Transformasi Fourier dan PCA," vol. 2, no. 1, 2021.
- [4] I. Perlindungan and Risnawati, "Pengenalan Tanaman Cabai dengan Teknik Klasifikasi Menggunakan Metode CNN," *Prosiding Seminar Nasional Mahasiswa Bidang Ilmu Komputer dan Aplikasinya*, vol. 1, no. 2, pp. 15–22, 2020, [Online]. Available: <https://conference.upnvj.ac.id/index.php/senamika/article/view/694>
- [5] Tinaliah and T. Elizabeth, "Penerapan Convolutional Neural Network Untuk Klasifikasi Citra Ekspresi Wajah Manusia Pada MMA Facial Expression Dataset," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 8, no. 4, pp. 2051–2058, 2021, [Online]. Available: <http://jurnal.mdp.ac.id>
- [6] D. Alamsyah and D. Pratama, "Implementasi Convolutional Neural Networks (CNN) Untuk Klasifikasi Ekspresi Citra Wajah Pada FER-2013 Dataset," *Jurnal Teknologi Informasi*, vol. 4, no. 2, pp. 350–355, 2020.
- [7] S. K. Ladi, G. K. Panda, R. Dash, P. K. Ladi, and R. Dhupar, "A Novel Grey Wolf Optimisation based CNN Classifier for Hyperspectral Image classification," *Multimed Tools Appl*, vol. 81, no. 20, pp. 28207–28230, 2022, doi: 10.1007/S11042-022-12628-2/METRICS.
- [8] V. Lestari, A. Kamsyakawuni, and K. A. Santoso, "Implementasi Algoritma Grey Wolf Optimizer (GWO) di Toko Citra Tani Jember," *Majalah Ilmiah Matematika dan Statistika*, vol. 19, pp. 65–74, 2019, [Online]. Available: <https://jurnal.unej.ac.id/index.php/MIMS/index>
- [9] A. Jamshed, B. Mallick, and R. K. Bharti, "An Efficient Pattern Mining Convolution Neural Network (CNN) Algorithm With Grey Wolf Optimization (GWO)," *Arxiv - Computer Vision and Pattern Recognition*, 2022, doi: <https://doi.org/10.48550/arXiv.2204.04704>.
- [10] Z. F. Abror, "Klasifikasi Citra Kebakaran dan Non Kebakaran Menggunakan Convolutional Neural Network," *Jurnal Ilmiah Teknologi dan Rekayasa*, vol. 24, no. 2, pp. 102–113, 2020, doi: 10.35760/TR.2019.V24I2.2389.
- [11] Suyanto, *Machine Learning : Tingkat Dasar Dan Lanjut*. Informatika, 2018.
- [12] Setyobudi, R. (2023). Utilization of tds sensors for water quality monitoring and water filtering of carp pools using IoT. *EUREKA: Physics and Engineering*, (6), 69-77.
- [13] Maradi, A. Y. (2020). Pemanfaatan android untuk sistem kendali robot penembak dengan mikrokontroler. *CYCLOTRON*, 3(1).
- [14] Sholihah, N., Fajri, F. N., & Khairi, M. (2022). Rekam Data Self Manajemen Cairan Pada Pasien Gagal Ginjal Kronis Yang Menjalani Terapi Hemodialisa Di RSUD Abdoer Rahem Berbasis Android. *Jurnal Manajemen Informasi Kesehatan Indonesia*, 10(1), 6-6.
- [15] Riadi, R. S., Rizal, S., Hamzah, S. H., Hidayatullah, S., & Rohim, A. (2022). Sistem Informasi Ulem-Ulem Berbasis Android Sebagai Upaya Pelestarian Kearifan Lokal Budaya Ulem-Ulem (Studi Kasus: Desa Jabung sisir Kecamatan Paiton Kabupaten Probolinggo).