

## IMPLEMENTASI ALGORITMA CNN DALAM MENGIDENTIFIKASI TINGKAT KEPARAHAN JERAWAT PADA WAJAH

Ekin Adhi Guna, Esra Fransiska Sihombing, Michael Nico Pasaribu,  
Hermawan Syahputra, Fanny Ramadhani

Ilmu Komputer, Universitas Negeri Medan

Jl. William Iskandar, Kec. Percut Sei Tuan, Kab. Deli Serdang, Sumatera Utara, Indonesia

ekinadhiguna@gmail.com

### ABSTRAK

Penggunaan teknologi komputer di sektor kesehatan telah berkembang pesat, terutama dalam sistem rekomendasi dan promosi kesehatan. Perkembangan kecerdasan buatan, khususnya *deep learning*, telah terbukti sangat bermanfaat dalam deteksi dini penyakit melalui analisis pencitraan medis. *Convolutional Neural Network* (CNN), sebagai salah satu bagian dari *deep learning*, menunjukkan kinerja tinggi dalam berbagai aplikasi, termasuk identifikasi jerawat. Jerawat pada remaja memiliki dampak signifikan pada aspek personal, sosial, dan psikologis, seringkali menyebabkan depresi dan rendahnya percaya diri. Pengetahuan masyarakat tentang perawatan jerawat yang tepat masih terbatas, yang sering kali menyebabkan kesalahan perawatan dan memperburuk kondisi. Penelitian ini bertujuan untuk mengembangkan sistem berbasis CNN guna mengidentifikasi tingkat keparahan jerawat sehingga dapat memberikan penanganan yang tepat. Implementasi CNN memungkinkan model untuk mengekstraksi hierarki fitur dari citra, sehingga menciptakan representasi visual yang lebih baik untuk variasi dari tingkat keparahan jerawat. Hasil penelitian dengan 1.106 dataset dari tiga label menunjukkan akurasi sebesar 75%, menegaskan efektivitas CNN dalam mengidentifikasi pola visual tingkat keparahan jerawat. Sistem ini diharapkan mampu memberikan solusi dalam mengatasi masalah keparahan jerawat dan mengurangi kesalahan perawatan yang sering terjadi, sehingga meningkatkan kualitas hidup remaja yang terpengaruh oleh jerawat.

**Kata kunci :** CNN, Python, Colab, Jerawat, Identifikasi

### 1. PENDAHULUAN

Sektor kesehatan kini mulai mengadopsi penggunaan komputer tidak hanya sebagai alat promosi dan pengajaran tetapi juga sebagai sistem rekomendasi. Kemajuan dalam teknologi kecerdasan buatan, khususnya melalui pengembangan *deep learning*, telah terbukti bermanfaat dalam membantu deteksi dini penyakit, terutama dalam aspek pencitraan medis. Teknologi telah memungkinkan komputer untuk mengidentifikasi pola, menganalisa data dan teks, serta menginterpretasi gambar. *Deep learning* khususnya, sangat efektif dalam mendeteksi objek. Proses ini, terutama dalam konteks gambar, sangat bergantung pada dukungan dari *Graphic Processing Unit* (GPU) untuk menghasilkan deteksi yang akurat [1]. Karena keefektifannya ini, *deep learning* menjadi sangat populer dalam pengolahan data kesehatan.

Salah satu bagian dari *deep learning* adalah *Convolutional Neural Network* (CNN) yang merupakan jenis *neural network* yang telah digunakan di berbagai bidang dengan kinerja yang tinggi. Kinerja yang tinggi ini memicu pengembangan aplikasi di bidang kesehatan kulit dan jerawat [2]. Jerawat pada remaja dapat memberikan dampak pada aspek personal-sosial dan psikologis. Remaja dengan jerawat dapat mengalami depresi, harga diri rendah, dan ketakutan untuk bersosialisasi. Seseorang yang memiliki jerawat mungkin merasa tidak nyaman dan memiliki perasaan buruk tentang diri mereka sendiri. Selain itu, konsep diri remaja dapat berubah jika

mereka memahami betapa pentingnya penampilan dalam interaksi sosial [3].

Jerawat adalah penyakit kulit yang sering terjadi, ditandai dengan munculnya komedo, papula, nodul, dan pustula. Biasanya, jerawat muncul di wajah, leher, bahu, dada, punggung, dan lengan atas. Penyakit ini sangat umum, menyerang 80% remaja laki-laki dan perempuan. Remaja laki-laki biasanya mengalaminya pada usia 16-19 tahun, sedangkan remaja perempuan pada usia 14-17 tahun. Jerawat tidak hanya menyerang remaja. Menurut studi *Golden Burden of Disease* (GBD), 85% orang dewasa muda usia 12-25 tahun juga mengalaminya. Penelitian di Jerman menunjukkan 64% orang berusia 20-29 tahun dan 43% orang berusia 30-39 tahun masih memiliki jerawat. Di Indonesia, berdasarkan catatan dermatologi kosmetik, persentase penderita jerawat meningkat dari 60% di tahun 2006 menjadi 80% di tahun 2007 dan 90% di tahun 2009 [4].

Masyarakat awam umumnya memiliki pengetahuan yang terbatas dalam merawat jerawat. Kesalahan dalam perawatan jerawat dapat memperparah kondisinya karena setiap tingkatan jerawat jerawat memerlukan penanganan yang berbeda, meskipun gejalanya hampir sama. Penelitian ini dilakukan untuk membuat sistem yang dapat mengidentifikasi tingkat keparahan dengan algoritma CNN dengan penanganan yang sesuai berdasarkan tingkat keparahan, risiko terjadinya bekas luka, hiperpigmentasi, dan kerusakan kulit lainnya dapat diminimalisir.

CNN merupakan jenis jaringan saraf yang dalam penggunaannya umumnya diterapkan pada data berupa gambar. Penerapan algoritma CNN pada gambar bertujuan untuk mengenali serta mendeteksi objek yang ada dalam gambar tersebut. Arsitektur CNN meliputi *input image*, *convolutional layer*, *pooling layer* dan *fully connected layer* [5].

Dengan menerapkan CNN dalam proses mengidentifikasi tingkat keparahan jerawat diharapkan dapat memberikan solusi terhadap permasalahan mengenai keparahan jerawat beserta kesalahan perawatan yang diderita. Oleh karena itu, akan dibuat suatu Sistem identifikasi tingkat keparahan jerawat pada wajah. Sistem ini merupakan aplikasi berbasis komputer yang menggunakan Metode CNN.

## 2. TINJAUAN PUSTAKA

### 2.1. Jerawat

Jerawat (*Acne vulgaris*) adalah penyakit kulit yang paling sering dikeluhkan oleh masyarakat, terutama remaja. Penyakit ini biasanya muncul di permukaan kulit wajah, leher, dada, dan punggung. Jerawat terjadi ketika kelenjar minyak di kulit menjadi terlalu aktif, menyebabkan pori-pori kulit tersumbat oleh penumpukan lemak yang berlebihan. Secara klinis, jerawat ditandai dengan munculnya komedo, papul, pustul, nodul, dan jaringan parut [6].

### 2.2. Convolutional Neural Network (CNN)

Sebuah jenis arsitektur jaringan saraf buatan yang dirancang secara khusus untuk menangani data yang memiliki struktur tertentu, seperti gambar dan data berbentuk grid. CNN sangat efektif untuk berbagai tugas seperti pengenalan gambar, deteksi objek, segmentasi gambar, dan lain sebagainya [7]. Arsitektur CNN terinspirasi oleh cara kerja sistem visual manusia dan hewan.

Inti dari CNN adalah penggunaan prinsip konvolusi, di mana kernel (filter) konvolusi dengan ukuran tertentu digeser melintasi gambar. Proses ini memungkinkan komputer untuk membentuk representasi baru dari informasi yang dihasilkan oleh filter saat diterapkan pada berbagai bagian gambar. Arsitektur umum Convolutional Neural Network meliputi tiga komponen utama: input, ekstraksi fitur, dan klasifikasi [8].

### 2.3. Python

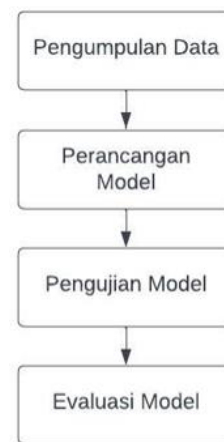
Python adalah *scripting language* yang berorientasi objek. Bahasa pemrograman ini dapat digunakan untuk pengembangan perangkat lunak dan bisa dijalankan melalui berbagai sistem operasi. Saat ini, Python juga merupakan bahasa yang populer bagi bidang *data science* dan analisis. Hal ini dikarenakan oleh dukungan bahasa Python terhadap library yang didalamnya menyediakan fungsi analisis data dan fungsi *machine learning*, *data preprocessing tools*, serta visualisasi data [9].

### 2.4. Identifikasi

Identifikasi adalah proses menetapkan identitas seseorang, benda, atau entitas lainnya. Secara umum, identifikasi merujuk pada tindakan memberikan tanda-tanda pada kelompok barang atau entitas tertentu, dengan maksud membedakan satu komponen dari yang lainnya, sehingga komponen tersebut dapat diakui dan diklasifikasikan ke dalam kelompok yang sesuai [10].

## 3. METODE PENELITIAN

Pada penelitian ini menerapkan tahapan penelitian seperti pada Gambar 1.



Gambar 1. Tahapan penelitian

### 3.1. Pengumpulan Data

Data yang digunakan untuk *training* dan *validation* adalah model klasifikasi tingkat keparahan jerawat yang dikumpulkan melalui kaggle. Data yang digunakan terdiri 1.106 foto yang terdiri atas 3 tingkatan (level) keparahan jerawat seperti pada Tabel 1.

Tabel 1. Dataset

Tingkatan	Jumlah Data	Data Train	Data Validation
Level 1	637	496	141
Level 2	186	135	51
Level 3	137	96	41

### 3.2. Perancangan Model

Pada tahapan ini dilakukan penentuan model CNN yang sesuai untuk mengklasifikasi gambar tingkat keparahan jerawat. CNN sendiri merupakan salah satu algoritma dari *deep learning* yang merupakan pengembangan dari *Multilayer Perceptron* (MLP) untuk mengolah data dalam bentuk *array matrix* berupa dua dimensi atau lebih, seperti data gambar, suara maupun video. CNN digunakan untuk mengklasifikasikan data berlabel menggunakan metode *supervised learning*. CNN sering digunakan untuk mengenali objek atau pemandangan, serta juga untuk melakukan deteksi dan segmentasi objek. Pada penelitian ini dilakukan implementasi arsitektur model *Convolutional Neural Network* (CNN) menggunakan

framework Keras dari TensorFlow yang digunakan untuk pemodelan klasifikasinya.

### 3.3. Pelatihan Model

Pada proses ini dilakukan pelatihan terhadap model dengan *dataset* yang telah disiapkan menggunakan metode *fit* dari keras dengan beberapa parameter yang telah digunakan. Proses pelatihan ini akan mengoptimalkan bobot-bobot model (*kernel* dan *bias*) untuk meminimalkan fungsi kerugian (*loss*) pada data *training*, dengan harapan bahwa model kemudian akan mampu menggeneralisasi dengan baik pada data yang belum pernah dilihat (*data testing*).

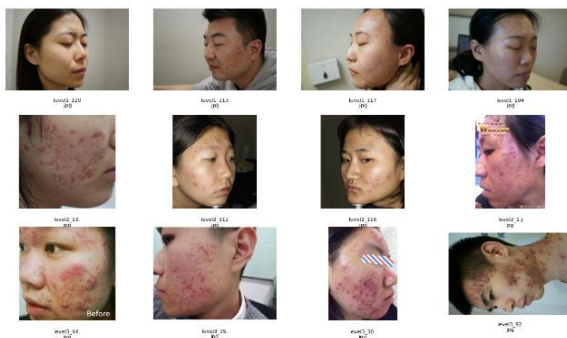
### 3.4. Evaluasi Model

Pada tahap ini dilakukan pengevaluasian model pada data *testing* setelah proses pelatihan selesai. Dengan menggunakan metode *evaluate* pada model untuk mengukur performa model pada data validasi akan memberikan *output* berupa nilai akurasi dan fungsi kerugian (*loss*). Pada tahap ini juga akan dibuat suatu *plot* untuk menunjukkan peforma model pada setiap tahapan pelatihan. Hasil evaluasi ini memberikan gambaran tentang seberapa baik model berperforma pada data yang tidak pernah dilihat selama proses pelatihan.

## 4. HASIL DAN PEMBAHASAN

### 4.1. Pengumpulan Data

Pada penelitian ini digunakan 3 level keparahan jerawat yang digunakan sebagai bahan untuk membuat klasifikasi menggunakan algoritma CNN. Pada Gambar 2 adalah sampel dari *dataset* yang digunakan dalam penelitian ini.



Gambar 2. Sampel dataset

### 4.2. Perancangan Model

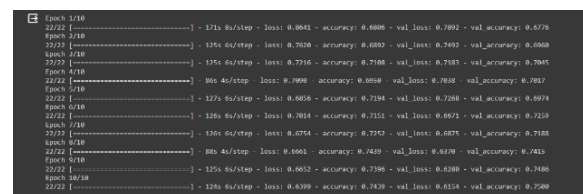
Dalam penelitian ini, *Convolutional Neural Network* (CNN) diimplementasikan menggunakan framework Keras dari TensorFlow untuk melakukan pemodelan klasifikasi. Langkah-langkah dalam merancang modelnya adalah sebagai berikut:

- Mengimport *library* yang dibutuhkan dalam pembuatan model klasifikasi.
- Membuat model *Sequential*, yang merupakan tumpukan linier dari lapisan-lapisan *neural network*.

- Menambahkan lapisan konvolusi dengan 32 filter, masing-masing berukuran 3x3, dan menggunakan fungsi aktivasi ReLU.
- Menambahkan lapisan *max pooling* dengan ukuran *pool 2x2*. *Max pooling* digunakan untuk mereduksi dimensi spasial dari representasi gambar.
- Menambahkan lapisan konvolusi kedua dengan 64 filter berukuran 3x3 dan aktivasi ReLU.
- Menambahkan lapisan *max pooling* kedua dengan ukuran *pool 2x2*.
- Menambahkan lapisan konvolusi ketiga dengan 128 filter berukuran 3x3 dan aktivasi ReLU.
- Menambahkan lapisan *max pooling* ketiga dengan ukuran *pool 2x2*.
- Meratakan (*flatten*) *output* dari lapisan sebelumnya menjadi vektor satu dimensi. Ini diperlukan sebelum memasukkan ke lapisan *Dense (fully connected)*.
- Menambahkan lapisan *fully connected* dengan 128 *neuron* dan fungsi aktivasi ReLU.
- Menambahkan lapisan *fully connected* terakhir dengan 3 *neuron*, yang sesuai dengan jumlah kelas yang diinginkan. Fungsi aktivasi *softmax* digunakan untuk menghasilkan distribusi probabilitas untuk setiap kelas.

### 4.3. Pelatihan Model

Pada tahap pelatihan model ini akan menghasilkan model CNN terlatih yang telah siap untuk diujikan. Proses pelatihan diiterasi sesuai dengan jumlah *epoch* yang ditentukan, di mana setiap *epoch* mewakili satu iterasi melalui seluruh data *training* untuk mengekstrak representasi fitur yang diperlukan. Dalam proses pelatihan, pengoptimal digunakan untuk memperbarui bobot selama fase *backward-pass*. Pengoptimal yang digunakan adalah pengoptimal Adam dengan *learning rate* 0.0001. Setelah proses pelatihan selesai, *accuracy* dan *loss value* yang diperoleh berdasarkan hasil setiap *epoch* akan tercatat pada Gambar 3.

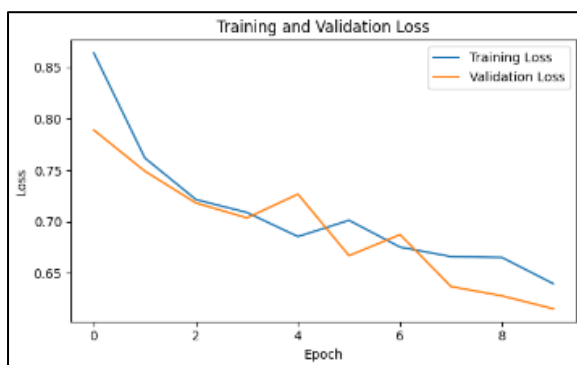


Gambar 3. Hasil pelatihan model

Setelah proses *epoch* selesai maka akan dibuat suatu plot dari semua hasil *epoch* proses *training* dan *validation* tersebut seperti yang ditampilkan pada Gambar 4 dan Gambar 5 untuk melihat peforma dari model yang telah dibuat. Dari visualisasi tersebut, dapat disimpulkan bahwa model menunjukkan kinerja cukup baik. Selama proses pelatihan, terjadi peningkatan yang stabil pada nilai akurasi dan konsistensi pada nilai kerugian (*loss value*), dengan *accuracy* mencapai 0.7500 dan kerugian 0.6154.



Gambar 4. Plot accuracy



Gambar 5. Plot loss value

#### 4.4. Evaluasi Model

Setelah dilakukan pelatihan model tahapan selanjutnya adalah evaluasi terhadap model tersebut. Dengan menggunakan metode *evaluate* pada model untuk mengukur performa model pada data *validation* maka diperoleh *loss value* sebesar 0.6153 atau 61% dan untuk *accuracy* sebesar 0.7483 atau 75%. Hasil evaluasi ini memberikan gambaran tentang seberapa baik performa model pada data yang tidak pernah dilihat selama proses pelatihan (*data validation*).

```
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
from tensorflow.keras.preprocessing import image

#Path gambar yang ingin diprediksi
img_path = 'content/drive/MyDrive/Colab Notebooks/keras_level/Validation/level 1/level_531.jpg'

#Membaca dan menampilkan gambar
img = mpimg.imread(img_path)
plt.figure(figsize=(10, 10))
plt.imshow(img)
plt.axis('off')
plt.show()

#Memproses gambar menggunakan model
img = image.load_img(img_path, target_size=(64, 64))
img_array = image.img_to_array(img)
img_array = np.expand_dims(img_array, axis=0)
img_array /= 255.0

predictions = model.predict(img_array)
class_index = np.argmax(predictions)







class_labels = ['level 1', 'level 2', 'level 3']
predicted_class = class_labels[class_index]
print(f'Predicted Class: {predicted_class}')
```

Gambar 6. Proses pengujian model

Gambar 6 menunjukkan tampilan dari proses pengujian model yang telah dilatih, termasuk beberapa tahapan seperti pemasukan library, penentuan path gambar, proses memuat dan menampilkan gambar, serta proses prediksi gambar. Proses ini dilakukan secara berulang untuk mengevaluasi dan menyimpulkan performa dari model yang telah dilatih.

Pada Tabel 2 memuat beberapa hasil dari pengujian model yang telah dilatih sebelumnya, data yang digunakan dalam pengujian ini adalah data dari dataset tingkat keparahan yang diambil secara acak tiap kelasnya. Berdasarkan tabel tersebut terlihat bahwa dari 6 pengujian menunjukkan hasil dimana terdapat 4 benar dan 2 salah, sehingga dapat dikatakan bahwa model dapat melakukan identifikasi tingkat keparahan jerawat dengan cukup baik.

Tabel 2. Hasil pengujian model

No	Gambar	Label Benar	Prediksi	Status
1		Level 1	Level 1	Benar
2		Level 1	Level 1	Benar
3		Level 2	Level 1	Salah
4		Level 2	Level 1	Salah
5		Level 3	Level 3	Benar
6		Level 3	Level 3	Benar

#### 5. KESIMPULAN DAN SARAN

Dari hasil pengujian yang dilakukan dalam penelitian ini, dapat disimpulkan bahwa penerapan algoritma *Convolutional Neural Network* (CNN) dalam mengidentifikasi tingkat keparahan jerawat berhasil. Penggunaan CNN memungkinkan model untuk secara progresif mengekstraksi hierarki fitur dari citra, sehingga menciptakan representasi yang lebih baik dari variasi visual yang ada pada berbagai tingkat keparahan jerawat. Pengujian menggunakan 1.106 dataset dari 3 label telah menghasilkan tingkat akurasi sebesar 75%, menunjukkan efektivitas CNN dalam mengidentifikasi pola visual pada tingkat keparahan jerawat. Implementasi ini berpotensi meningkatkan efisiensi dalam proses klasifikasi tingkat keparahan jerawat dengan akurasi yang sudah cukup baik, memberikan kontribusi penting dalam pengembangan sistem pengenalan objek di bidang pengolahan citra digital. Secara keseluruhan, penelitian ini memperkuat peran algoritma CNN sebagai solusi yang cukup efektif untuk klasifikasi tingkat keparahan jerawat.

## DAFTAR PUSTAKA

- [1] E. Naf'an, F. Islami, and G. Gushelmi, "Implementasi Deep Learning Dalam Pendeteksian Kerumunan Yang Berpotensi Melanggar Protokol Kesehatan Covid-19," *J. Media Inform. Budidarma*, vol. 6, no. 2, p. 821, 2022, doi: 10.30865/mib.v6i2.3484.
- [2] S. Rekomendasi pada Forum Kesehatan Dengan Pemeringkatan Pertanyaan Serupa Menggunakan Pendekatan Deep Learning, B. Khufa Rahmada Aula, C. Fatichah, and D. Purwitasari, "The Journal on Machine Learning and Computational Intelligence (JMLCI) 28," pp. 28–33, [Online]. Available: [www.dokter.id](http://www.dokter.id)
- [3] Aryani Diah Tri and Rianingrum Wahyu, "Hubungan Acne Vulgaris (AV) Dengan Kepercayaan Diri Pada Mahasiswa Universitas Purwokerto Angkatan 2021," *J. Kesehat. Tambusai*, vol. 3, no. 3, pp. 2774–0524, 2022.
- [4] Y. F. Achmad, A. Yulfitri, and M. B. Ulum, "Identifikasi Jenis Jerawat Berdasarkan Tekstur Menggunakan GLCM dan Backpropagation," *J. SAINTIKOM (Jurnal Sains Manaj. Inform. dan Komputer)*, vol. 20, no. 2, p. 139, 2021, doi: 10.53513/jis.v20i2.4747.
- [5] C. Oktaviany and T. L. Marselino, "Pengembangan Perangkat Lunak Untuk Pengelompokkan Tumbuhan Berdasarkan Citra Digital Daun Menggunakan CNN," *KALBISIANA J. Sains, Bisnis dan ...*, vol. 8, no. 2, pp. 1895–1914, 2020, [Online]. Available: <http://112.78.142.42/index.php/kalbisiana/article/view/429%0Ahttp://112.78.142.42/index.php/kalbisiana/article/download/429/342>
- [6] I. Loni, Beta Ria Erika Marita Dellima, and Eni Kartika Sari, "UJI EFEKTIVITAS SEDIAAN GEL EKSTRAK ANGUR LAUT (*Caulerpa racemosa*) TERHADAP BAKTERI *Staphylococcus aureus* PENYEBAB JERAWAT," *J. Farm. Dan Kesehat. Indones.*, vol. 3, no. 2, pp. 114–125, 2023, doi: 10.61179/jfki.v3i2.471.
- [7] N. K. Qudsi, R. A. Asmara, and A. R. Syulistyo, "Identifikasi Citra Tulisan Tangan Digital Menggunakan Convolutional Neural Network (CNN)," *Semin. Inform. Apl. Polinema*, pp. 48–53, 2019.
- [8] R. Obet Yumame, Y. Agus Pranoto, and J. Dedy Irawan, "Rancang Bangun Pendeteksi Masker Dan Alat Pembersih Tangan Otomatis Berbasis Arduino Menggunakan Metode Cnn (Convolution Neural Network) Di Gereja Gki Jayapura Papua," *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 6, no. 2, pp. 965–973, 2023, doi: 10.36040/jati.v6i2.5408.
- [9] N. H. Harani, C. Prianto, and M. Hasanah, "Deteksi Objek Dan Pengenalan Karakter Plat Nomor Kendaraan Indonesia Menggunakan Metode Convolutional Neural Network (CNN) Berbasis Python," *J. Tek. Inform.*, vol. 11, no. 3, pp. 47–53, 2019, [Online]. Available: <https://ejurnal.ulbi.ac.id/index.php/informatika/article/view/658>
- [10] R. Pujiati and N. Rochmawati, "Identifikasi Citra Daun Tanaman Herbal Menggunakan Metode Convolutional Neural Network (CNN)," *J. Informatics Comput. Sci.*, vol. 3, no. 03, pp. 351–357, 2022, doi: 10.26740/jinacs.v3n03.p351-357.