

Komparasi Algoritma MLP+LBP dan CNN Sebagai solusi Inovatif Untuk Deteksi Dini Korosi

Diterima:
10 Juni 2024
Revisi:
10 Juli 2024
Terbit:
1 Agustus 2024

^{1*}Mizanul Ridho Aohana [0009-0006-2228-6718], ²Fitri Bimantoro, ³Ratu Nisful Laily Hidhayah, ⁴Daniel Swanjaya
¹⁻³Universitas Mataram, ⁴Universitas Nusantara PGRI Kediri
¹mizanul.aohana@gmail.com, ²bimo@unram.ac.id,
³ratunisful16@gmail.com, ⁴daniel@unpkediri.ac.id

Abstrak— Logam dan bahan lain seperti logam dapat mengakibatkan kerugian finansial besar dan bahaya bagi keselamatan umum terutama pekerja di dunia industri seperti industri perkapalan, tambang dan logam. Penelitian ini bertujuan untuk meningkatkan efektivitas deteksi dini korosi menggunakan metode *Convolutional Neural Network* (CNN) dibandingkan dengan penelitian sebelumnya yang menggunakan metode *Multi-Layer Perceptron* (MLP) dikombinasikan dengan *Local Binary Pattern* (LBP) sebagai teknik ekstraksi fitur. Dataset pada penelitian ini terdiri dari 1630 gambar korosi dan non-korosi yang diperoleh melalui scraping Google Image. Model CNN yang diusulkan menunjukkan akurasi deteksi sebesar 85,16% dan *Area Under Curve* (AUC) sebesar 91,63%. Ini lebih baik daripada MLP+LBP, yang menunjukkan akurasi 84,92% dan AUC 91,52%. Meskipun perbedaan ini relatif kecil, hal menarik terletak pada jumlah parameter yang digunakan oleh CNN lebih kecil (1.594.337) jika dibandingkan dengan MLP+LBP (6.558.650). Hasil ini menunjukkan bahwa secara keseluruhan, CNN lebih baik dan efisien dalam mendeteksi korosi.

Kata Kunci— Korosi, CNN, MLP, LBP, Ekstraksi Fitur, Deteksi Otomatis.

Abstract— *Corrosion of metals and other metallic materials can lead to significant financial losses and pose a serious threat to public safety, particularly for workers in industrial sectors such as shipbuilding, mining, and metallurgy.. This research aims to enhance the effectiveness of early corrosion detection employing the Convolutional Neural Network (CNN) method compared to previous studies that utilized the Multi-Layer Perceptron (MLP) method in conjunction with Local Binary Pattern (LBP) as a feature extraction technique. The dataset in this research consists of 1630 corrosion and non-corrosion images obtained through scraping Google Images. The proposed CNN model shows a detection accuracy of 85.16% and an Area Under Curve (AUC) of 91.63%. This is better than MLP+LBP, which showed an accuracy of 84.92% and an AUC of 91.52%. While the difference is relatively small, the interesting point lies in the number of parameters used by CNN, which is smaller (1,594,337) compared to MLP+LBP (6,558,650). This result indicates that overall, CNN is better and more efficient in detecting corrosion.*

Keywords— *Corrosion, CNN, MLP, LBP, Feature Extraction, Automatic Detection.*

This is an open access article under the CC BY-SA License.



Penulis Korespondensi:

Mizanul Ridho Aohana
Program Studi Teknik Informatika, Fakultas Teknik
Universitas Mataram
Email: mizanul.aohana@gmail.com
ID Orcid: [0009-0006-2228-6718]
Handphone: 081915634615

I. PENDAHULUAN

Korosi menjadi masalah yang sering ditemui di berbagai material seperti logam [1], korosi dapat menimbulkan kerugian yang signifikan yang dapat mempengaruhi umur dan kinerja material, kerugian yang pernah terjadi mencapai 2,5 triliun rupiah akibat kebocoran pipa PAM Jaya di Jakarta, kerugian ini membesar diakibatkan kebocoran pipa yang mencapai 46% salah satu penyebabnya adalah pipa-pipa yang tertanam sudah berusia tua dan mengalami korosi. Kerusakan akibat korosi tidak hanya mempengaruhi industri, tetapi juga dapat memiliki dampak yang serius pada keselamatan publik. Sebuah penelitian menyebutkan bahwa korosi pada kaleng makanan dapat menyebabkan kontaminasi lapisan timah putih dalam kaleng yang terkelupas, yang berpotensi membahayakan kesehatan konsumen [2].

Oleh karena itu diperlukan cara untuk mencegah terjadinya Korosi. Pencegahan yang umum dilakukan adalah memberikan lapisan cat anti korosi. Menurut hasil uji probabilitas korosi, tulangan baja yang menggunakan cat anti-korosi berbasis bitumen memiliki nilai potensi korosi lebih rendah daripada spesimen yang tidak menerima lapisan [3], disertai dengan pemantauan rutin [2]. Sehingga masalah korosi dapat diturunkan, untuk meningkatkan umur dan kinerja material agar lebih lama, serta mengurangi biaya perbaikan dan penggantian yang tidak diinginkan [4].

Pencegahan karat dengan gaya konvensional (pemeriksaan manual) memerlukan waktu yang lama dan kurang efisien, selain waktu yang lama, resiko keselamatan kerja juga sering terjadi akibat permasalahan karat. Oleh karena itu diperlukan suatu sistem yang dapat mengidentifikasi sebuah karat dengan cepat dan akurat [5]. Perkembangan *machine learning* saat ini memungkinkan pendeteksian korosi secara otomatis dengan akurasi dan efektivitas lebih tinggi [6], [7]. Salah satu metode *machine learning* yang memiliki peran dalam meningkatkan pendeteksian korosi melalui analisis gambar adalah jaringan syaraf tiruan melalui metode CNN [7], [8].

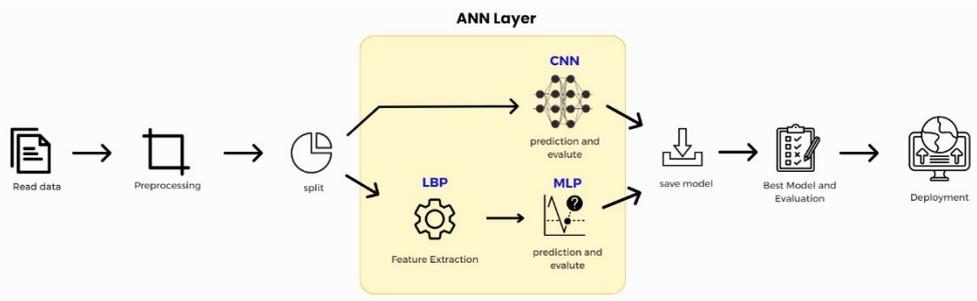
Jika dilihat pada penelitian sebelumnya, metode CNN (*Convolutional Neural Network*) yang dapat mendeteksi gambar dari piksel ke piksel, dianggap sebagai metode yang efektif untuk menemukan penyakit dalam gambar tanaman padi. Dataset yang digunakan dalam penelitian ini terdiri dari 1630 data, yang dibagi menjadi tiga kelas penyakit. Selain itu, dalam penelitian tersebut menggunakan arsitektur CNN *InceptionV3* dan melakukan perbandingan jumlah epoch. Hasil penelitian menunjukkan kinerja yang sangat baik—dengan akurasi 98% tanpa *overfitting*—bahwa teknik yang digunakan dapat mendeteksi penyakit pada daun tanaman padi dengan hasil yang memuaskan [9].

Penelitian ini merupakan penelitian lanjutan dari penelitian sebelumnya yang membandingkan beberapa metode ekstraksi fitur dalam pendeteksian karat [10]. Namun, analisis gambar untuk pendeteksian korosi dapat ditingkatkan lebih jauh dengan pengembangan algoritma *deep learning* (CNN). Sebagai pembanding, penggunaan algoritma CNN dalam pendeteksian korosi pada penelitian ini diharapkan mampu memberikan akurasi yang lebih tinggi dibandingkan dengan metode sebelumnya seperti MLP dan LBP, serta meningkatkan efisiensi dan kecepatan identifikasi karat.

II. METODE

A. Proposed Work

Peneliti mengusulkan tahap proses yang dimulai dengan membaca data dan kemudian melakukan *preprocessing*. Untuk mempercepat pemrosesan data, dataset di-*resize* menjadi 256 x 256 dan kemudian dibagi menjadi beberapa folder untuk diproses dengan berbagai metode, seperti yang ditunjukkan pada Gambar 1. Model yang dihasilkan dari proses ini kemudian disimpan dan dapat digunakan untuk tujuan lain.



Gambar 1. Alur Penelitian

B. Multi-Layer Perceptron (MLP)

MLP merupakan salah satu arsitektur *Artificial Neural Network* paling populer, MLP dianggap sebagai yang pertama menggunakan algoritma pelatihan *backpropagation* untuk MLP. Kebanyakan jaringan multilayer menggunakan fungsi aktivasi sigmoid di lapisan tersembunyi. Karena mereka memampatkan rentang masukan tak terbatas menjadi rentang keluaran terbatas, fungsi-fungsi ini terkadang disebut sebagai fungsi "*squashing*".

Kemiringan yang meningkat seiring dengan peningkatan masukan menunjukkan fungsi sigmoid. Hal ini menyebabkan masalah saat melatih jaringan multilayer dengan fungsi sigmoid dengan penurunan paling curam. Ini karena ukuran gradien yang sangat kecil dapat menyebabkan bias dan perubahan bobot yang kecil [3], [11].

Arsitektur MLP-ANN terdiri dari sekumpulan *node* yang saling berhubungan yang terdiri dari tiga jenis lapisan: *input*, *hidden layer*, dan *output*. Berikut penjelasan setiap *layer*:

a. Input layer

Input Layer adalah lapisan yang menerima nilai *input* dari setiap daftar dalam data Anda. Jumlah *node input* sebanding dengan jumlah variabel prediktor.

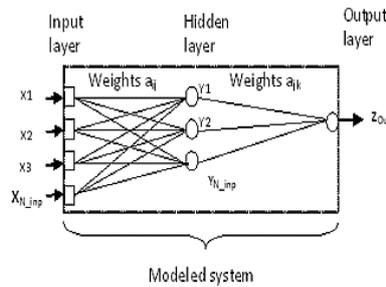
b. Hidden Layer

Nilai *input* jaringan diubah oleh *hidden layer*. Setiap *node* pada lapisan tersembunyi terhubung ke *node* lapisan tersembunyi sebelumnya atau dari *node* lapisan masukan, dan kemudian terhubung ke *node* lapisan tersembunyi berikutnya atau dari *node* lapisan keluaran. Ada banyak lapisan tersembunyi yang dapat Anda capai.

c. Output Layer

Nilai keluaran *output layer* berasal dari lapisan masukan atau lapisan tersembunyi dan biasanya berupa nilai floating point antara 0 dan 1 [12].

Ilustrasi komponen MLP yang berbeda dengan lapisan tersembunyi. Lapisan masukan hanya mengirimkan nilai x_i ke unit-unit yang ada pada lapisan tersembunyi dan tidak melakukan apa-apa terhadap sinyal masukan. Sinyal diterima oleh *node* lapisan masukan dan diubah menjadi sinyal, yang kemudian dikirim ke semua *node* keluaran dan diubah menjadi keluaran. Algoritma penurunan paling curam digunakan untuk mengubah berat koneksi dari masukan ke *node* lapisan tersembunyi dan dari lapisan tersembunyi ke lapisan keluaran. Metode ini dapat diselesaikan dengan masalah non-linier [12].



Gambar 2. Arsitektur *Multi-Layer Perceptron*

Multi-Layer Perceptron, juga dikenal sebagai MLP, adalah salah satu model pembelajaran supervisi *Artificial Neural Network* (ANN), yang memiliki jumlah neuron yang dapat disesuaikan sesuai kebutuhan. *Multi-Layer Perceptron* digunakan untuk melatih fitur korosi dalam penelitian ini.

Kemampuan MLP untuk menangani tugas prediksi yang rumit dan belajar dari pola data yang kompleks adalah alasan algoritma ini dipilih. Dalam situasi ini, ekstraksi fitur dengan menggunakan metode *Gray Level Co-occurrence Matrix* (GLCM), *Gray Level Run Length Matrix* (GLRLM), dan *Local Binary Pattern* (LBP) memberikan gambaran yang kaya dan menarik tentang struktur tekstur gambar korosi.

MLP memiliki keunggulan dalam mengekstraksi dan memodelkan pola yang kompleks dari data bertipe gambar, sehingga dapat secara efektif memanfaatkan informasi yang dihasilkan oleh metode ekstraksi fitur seperti GLRLM, GLCM CNN, dan LBP. Dengan kemampuan ini, MLP dapat memahami hubungan antara fitur ekstraksi dan tingkat korosi yang terkait, yang memungkinkan untuk melakukan prediksi korosi dengan tingkat akurasi yang tinggi dengan menggunakan MLP untuk melatih model.

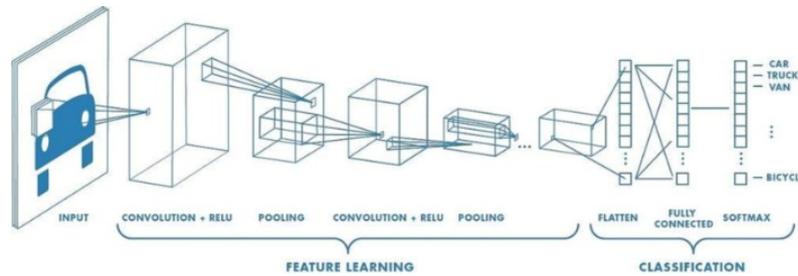
C. *Convolutional Neural Network* (CNN)

Salah satu pendekatan *deep learning* yang menggunakan jaringan syaraf tiruan untuk memecahkan masalah dengan mempelajari data yang ada adalah CNN. Metode CNN merupakan salah satu metode terbaik untuk pengenalan citra digital karena didasarkan pada sistem pengenalan citra pada *visual cortex* manusia [7].

Proses utama yang mendasari sebuah CNN adalah lapisan konvolusi, yang melakukan operasi konvolusi pada *output* lapisan sebelumnya. Lapisan konvolusi adalah metode yang paling umum digunakan dalam pengolahan citra. CNN adalah evolusi dari metode *Multi-layer Perceptron* (MLP) dan merupakan salah satu algoritma *deep learning*. Istilah matematis yang disebut "konvolusi" mengacu pada penerapan fungsi pada output fungsi lain berulang kali [13].

Tujuan dari lapisan penuh terhubung ini, yang biasanya digunakan dalam implementasi MLP, adalah untuk mengubah dimensi data sehingga data dapat diklasifikasikan secara linear.

Setiap neuron pada lapisan konvolusi harus diubah menjadi data satu dimensi sebelum dapat dimasukkan ke dalam lapisan penuh terhubung [7].



Gambar 3. Convolutional Neural Network (CNN)

D. Metrik Evaluasi

Matrix evaluasi digunakan untuk mengukur keakuratan model yang diperoleh, salah satu yang paling sering digunakan adalah *confusion matrix*. Tabel *confusion matrix* digunakan untuk mengevaluasi kinerja suatu model klasifikasi pada pembelajaran mesin. *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN) adalah empat nilai evaluasi utama dalam *confusion matrix*. Jika diturunkan lebih lanjut maka akan diperoleh beberapa metrik utama, dalam penelitian ini akan digunakan 3 metrik utama, yakni *accuracy*, *precision*, dan *recall*.

Accuracy merupakan rasio prediksi benar (positif dan negatif) yang diukur dengan keseluruhan data. Tujuannya adalah untuk mengetahui seberapa sering model pembelajaran mesin memprediksi hasil dengan tepat.

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \dots \dots \dots (1)$$

Recall akan mengukur tingkat *True Positive* dari hasil prediksi, untuk menghitung nilai tersebut, dinyatakan dalam persamaan 2.

$$Recall = \frac{TP}{TP + FN} \dots \dots \dots (2)$$

Precision memberikan nilai tingkat positif yang sebenarnya dari hasil prediksi, persamaannya dinyatakan dalam persamaan 3.

$$Precision = \frac{TP}{TP + FP} \dots \dots \dots (3)$$

ROC atau *Receiver Operating Characteristic* merupakan sebuah grafik yang mengilustrasikan kemampuan diagnostik dari sistem pengklasifikasi biner karena variasi ambang diskriminasi. dinyatakan dalam persamaan 4.

$$ROC = \frac{P(x \vee positive)}{P(x \vee negative)} \dots \dots \dots (4)$$

Area Under Curve (AUC) dihitung dengan menggabungkan luas trapesium yang dibentuk oleh titik-titik yang menunjukkan sensitivitas dan spesifisitas. Nilai AUC selalu berkisar antara 0 dan 1. Untuk mengevaluasi kinerja diskriminatif, metrik ini memperkirakan kemungkinan model untuk mengklasifikasikan sampel populasi yang dipilih secara acak menjadi negatif atau positif.

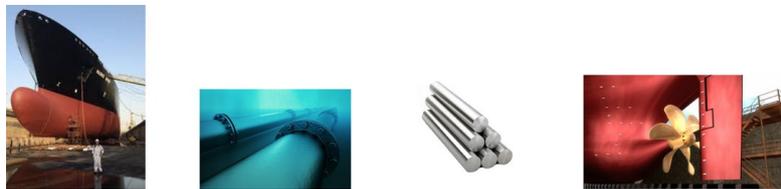
AUC pada dasarnya merupakan ukuran kemampuan model untuk menyusun peringkat instansi positif lebih banyak daripada instansi negatif. Ini berfungsi sebagai indikator akurasi model prediksi dan dihitung dengan melihat seberapa baik model dapat membedakan antara kelas. Nilai AUC berkisar antara 0 dan 1, dan nilai yang lebih tinggi menunjukkan klasifikasi yang lebih kuat [14].

Dalam penelitian ini, sama seperti penelitian sebelumnya, penulis akan menggunakan *accuracy*, *precision*, *recall* dan ROC/AUC sebagai metrik evaluasi algoritma yang akan diuji [15].

III. HASIL DAN PEMBAHASAN

A. Dataset

Dataset pada penelitian ini menggunakan dataset yang sama dengan penelitian yang sudah dilakukan sebelumnya. Dimana, dataset diperoleh melalui proses scraping dari Google Image dengan kata kunci "karat". Dataset ini terbagi ke dalam dua kelas, yaitu korosi dan non-korosi, dengan jumlah gambar secara berturut-turut sebanyak 990 gambar untuk kelas korosi dan 829 gambar untuk kelas non-korosi. Contoh dataset dapat dilihat pada Gambar 4 dan Gambar 5.



Gambar 4. Dataset Non-Korosi



Gambar 5. Dataset Korosi

Penggunaan dataset yang sama bertujuan untuk menjaga aspek kesetaraan dalam perbandingan hasil yang akan dilakukan dengan penelitian sebelumnya. Salah satu kelebihan dataset ini adalah tingginya variasi data atau objek yang ada sehingga dapat memberikan keuntungan pada proses pelatihan model. Dengan adanya variasi data ini, diharapkan model mampu memahami karakteristik karat dengan lebih baik.

B. Analisis dan Pembahasan

Pada penelitian ini, mekanisme pelatihan model juga menggunakan konsep *Multi-Layer Perceptron* dengan metode ekstraksi fitur *Local Binary Pattern* sebagai pembanding performa untuk algoritma CNN. Setelah melakukan beberapa percobaan arsitektur, diperoleh 3 hasil tertinggi secara berturut-urut sebagai berikut:

Tabel 1. Hasil Percobaan MLP+LBP

No	Arsitektur	Params.	Evaluasi			
			Acc	P	R	AUC
1	Number of Layers: 4, Number of Neurons in Each Layer: (100, 50), Solver: adam	6558650	84.92%	86.32%	85.42%	91.52%
2	Number of Layers: 4, Number of Neurons in Each Layer: (200, 50), Solver: adam	13117250	85.20%	83.90%	89.58%	91.51%
3	Number of Layers: 4, Number of Neurons in Each Layer: (50, 50), Solver: adam	3279350	85.20%	85.28%	87.50%	90.39%

Berdasarkan tabel 1, dapat dilihat bahwa performa terbaik diperoleh oleh arsitektur pertama dengan nilai akurasi sebesar 84,92 dan AUC sebesar 91.52%. Walaupun arsitektur kedua dan ketiga memiliki nilai akurasi yang lebih tinggi dari arsitektur pertama, nilai AUC yang diberikan oleh kedua arsitektur ini relatif lebih rendah dari arsitektur pertama. Namun, jika dilihat lebih detail kembali ternyata nilai AUC antara arsitektur 1 dan 2 hanya terpatut sekitar 0.01 %. Oleh karena itu dipertimbangkan pula efisiensi yang diberikan oleh masing-masing model dengan melihat jumlah parameter sehingga arsitektur 1 dipilih sebagai arsitektur yang memiliki performa paling ideal.

Berbeda dengan metode MLP+LBP yang memisahkan antara ekstraksi fitur dan pelatihan model, perbedaan konsep algoritma CNN menjadi menarik untuk dibahas karena mengusung penyatuan antara proses ekstraksi fitur dan training model. Setelah melakukan beberapa skema percobaan, diperoleh 3 hasil terbaik sebagai berikut:

Tabel 2. Hasil Percobaan CNN

No	Arsitektur	Params.	Epoch	Evaluasi			
				Acc	P	R	AUC
1	6 Conv, 6 Max Pool, 5 dropout, Flatten, 2 Dense	807905	30	82.97%	78.81%	93.94%	91.47%
2	6 Conv, 6 Max Pool, 5 dropout, Flatten, 2 Dense, padding=same	1594337	50	85.16%	81.03%	94.95%	91.63%
3	6 Conv, 6 Max Pool, 5 dropout, Flatten, 2 Dense, padding=same	1594337	75	83.52%	80.53%	91.92%	91.88%

Pada tabel 2, terlihat bahwa hasil terbaik diperoleh pada arsitektur kedua dengan nilai akurasi sebesar 85.16% dan nilai AUC sebesar 91.63%. Jika dibandingkan dengan 2 arsitektur lainnya, arsitektur kedua memiliki nilai yang lebih baik secara keseluruhan walaupun memiliki nilai AUC yang lebih rendah dari arsitektur ketiga.

Dalam kedua percobaan, hasil metrik evaluasi menunjukkan bahwa performa masing-masing algoritma tidak terpaut terlalu jauh. Tetapi jika diperhatikan lebih detail, perbedaan yang cukup mencolok terletak pada nilai *precision* dan *recall* kedua algoritma. Pada, algoritma MLP+LBP nilai *precision* dan *recall* yang dimiliki lebih seimbang jika dibandingkan dengan algoritma CNN. Hal ini kemungkinan besar disebabkan oleh arsitektur CNN yang dibangun belum mencapai titik optimal dan perlu dilakukan pengembangan lanjutan. Namun, hal lain yang menarik dalam penelitian ini adalah jumlah efisiensi parameter yang diberikan oleh algoritma CNN ternyata jauh lebih baik daripada menggunakan algoritma MLP+LBP.

IV. KESIMPULAN

Berdasarkan pembahasan yang sudah dipaparkan, hasil terbaik diperoleh oleh algoritma CNN dengan nilai *validation accuracy* sebesar 85.16% dan AUC sebesar 91.63%. Jika dibandingkan dengan algoritma MLP+LBP yang memiliki *validation accuracy* sebesar 84.92% dan AUC sebesar 91.52%, hasil ini sebenarnya tidak terlampau jauh. Selisih kedua algoritma ini hanya berkisar 0.24 % untuk *accuracy* dan 0.11 % untuk nilai AUC. Namun, hal yang menarik justru terletak pada jumlah parameter yang digunakan pada kedua algoritma ini. Pada algoritma CNN parameter yang digunakan hanya sebanyak 1.594.337, sedangkan algoritma MLP+LBP menggunakan parameter sebanyak 6.558.650, hampir 4x lebih banyak dari algoritma CNN. Dengan hasil ini, maka dapat disimpulkan bahwa algoritma CNN mampu menangani permasalahan karat dengan lebih baik dan efisien jika dibandingkan dengan algoritma MLP+LBP.

DAFTAR PUSTAKA

- [1] D. Wardani et al., "Penggunaan Metode Image Processing Sebagai Alat Karakterisasi Hasil Pelapisan pada Lambung Kapal," *Journal of Applied Electrical Engineering*, vol. 7, pp. 37–41, Jun. 2023, doi: 10.30871/jaee.v7i1.5435.
- [2] A. Affandi et al., "Analisa Korosi Atmosferik Baja Karbon Rendah Di Kecamatan Medan Belawan," *MULTITEK INDONESIA*, vol. 14, no. 2, pp. 80–88, Dec. 2020, doi: 10.24269/mtkind.v14i2.2841.
- [3] M. H. Santoso, D. A. Larasati, and Muhathir, "Wayang Image Classification Using MLP Method and GLCM Feature Extraction," *Journal of Computer Science, Information Technologi and Telecommunication Engineering*, vol. 1, no. 2, pp. 111–119, Sep. 2020, doi: 10.30596/jcositte.v1i2.5131.
- [4] M. Sofian et al., "Perlindungan Korosi Di Perkapalan," *Jurnal Pendidikan Teknik Mesin*, vol. 22, no. 2, pp. 50–56, Dec. 2022, doi: <https://doi.org/10.15294/jptm.v22i2>.
- [5] A. Royani, M. Hanafi, H. Julistiono, and A. Manaf, "Biokorosi Dan Teknologi Pencegahannya Di Industri Minyak Dan Gas," *Metalurgi*, vol. 36, no. 3, p. 135, Feb. 2022, doi: 10.14203/metalurgi.v36i3.608.
- [6] T. Sutojo, S. Rustad, M. Akrom, A. Syukur, G. F. Shidik, and H. K. Dipojono, "A Machine Learning Approach for Corrosion Small Datasets," *Npj Mater Degrad*, vol. 7, no. 1, p. 18, Mar. 2023, doi: 10.1038/s41529-023-00336-7.
- [7] B. Y. E. Purba, N. F. Saragih, A. P. Silalahi, S. Sitepu, and A. Gea, "Perancangan Alat Pendeteksi Kematangan Buah Nanas Dengan Menggunakan Mikrokontroler Dengan Metode Convolutional Neural Network (CNN)," *Methodika : Jurnal Ilmiah Teknik Informatika*, vol. 2, no. 1, pp. 13–21, Apr. 2022, [Online]. Available: <http://ojs.fikom-methodist.net/index.php/METHODIKA>
- [8] U. P. Sanjaya, Z. Alawi, A. R. Zayn, and G. P. Dirgantoro, "Optimasi Convolutional Neural Network dengan Standard Deviasi untuk Klasifikasi Pneumonia pada Citra X-rays Paru," *Generation Journal*, vol. 7, no. 3, pp. 40–47, Oct. 2023, doi: 10.29407/gj.v7i3.20183.

- [9] S. Sucipto, "Analisa Hasil Rekomendasi Pembimbing Menggunakan Multi-Attribute Dengan Metode Weighted Product," *Fountain of Informatics Journal*, vol. 2, no. 1, p. 27, May 2017, doi: 10.21111/fij.v2i1.912.
- [10] M. R. Aohana, R. N. L. Hidayah, M. J. Andara, N. Amara, and F. Bimantoro, "Review Komprehensif: Ekstraksi Fitur GLCM, GLRLM, dan LBP untuk Pendeteksian Korosi," *Seminar Nasional Teknologi & Sains*, vol. 3, no. 1, pp. 82–90, Jan. 2024, doi: 10.29407/stains.v3i1.4352.
- [11] A. Setiadi, "Penerapan Algoritma Multilayer Perceptron Untuk Deteksi Dini Penyakit Diabetes," *PARADIGMA: Jurnal Komputer dan Informatika Akademi Bina Sarana Informatika*, vol. 14, no. 1, pp. 46–59, Mar. 2012, doi: <https://doi.org/10.31294/p.v14i1.3378>.
- [12] M. Hasnain, M. F. Pasha, I. Ghani, M. Imran, M. Y. Alzahrani, and R. Budiarto, "Evaluating Trust Prediction and Confusion Matrix Measures for Web Services Ranking," *IEEE Access*, vol. 8, pp. 90847–90861, 2020, doi: 10.1109/ACCESS.2020.2994222.
- [13] H. Kusumah and R. A. Pradana, "Penerapan Trainer Interfacing Mikrokontroler Dan Internet Of Things Berbasis ESP32 Pada Mata Kuliah Interfacing," *Journal CERITA*, vol. 5, no. 2, pp. 120–134, Aug. 2019, doi: 10.33050/cerita.v5i2.237.
- [14] T. T. Maskoen and D. Purnama, "Area Under the Curve dan Akurasi Cystatin C untuk Diagnosis Acute Kidney Injury pada Pasien Politrauma," *Majalah Kedokteran Bandung*, vol. 50, no. 4, pp. 259–264, Dec. 2018, doi: 10.15395/mkb.v50n4.1342.
- [15] D. A. Setiawan, H. Fitriyah, and W. Kurniawan, "Sistem Klasifikasi Jenis Karat Menggunakan Metode Decision Tree Berbasis Raspberry Pi," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 3, no. 2, pp. 2114–2120, Jun. 2018, [Online]. Available: <https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/4479>