

IDENTIFIKASI GEJALA MONKEYPOX PADA KULIT MANUSIA MENGUNAKAN ARSITEKTUR EFFICIENTNET-B1 DENGAN OPTIMALISASI HYPERPARAMETER

Jessica Vega Nainggolan^{*1}, Subektiningsih²

^{1,2} Universitas Amikom Yogyakarta, Yogyakarta

Email: ¹jessica.vegaa@students.amikom.ac.id, ²subektiningsih@amikom.ac.id

^{*}Penulis Korespondensi

(Naskah masuk: 8 Februari 2025, diterima untuk diterbitkan: 29 Oktober 2025)

Abstrak

Monkeypox merupakan penyakit *zoonosis* yang memerlukan deteksi dini untuk penanganan yang tepat. Penelitian ini merancang model identifikasi gejala Monkeypox pada kulit manusia menggunakan arsitektur EfficientNet-B1 dengan optimalisasi *hyperparameter*. Dataset yang digunakan terdiri dari 3.192 gambar yang terbagi menjadi dua kelas, yaitu Monkeypox dan Non-Monkeypox. Proses perancangan model melibatkan tahapan identifikasi masalah, pengumpulan data, *pre-processing data*, perancangan arsitektur model dan pengimplementasian *hyperparameter*, evaluasi model, hingga analisis hasil. Evaluasi performa dilakukan dengan membandingkan model yang menggunakan optimalisasi *hyperparameter* dengan model yang tidak dioptimalisasi, menggunakan *confusion matrix*. Hasil penelitian menunjukkan bahwa model dengan optimalisasi *hyperparameter* mencapai performa yang sangat baik dengan nilai akurasi 99.22%, *precision* 99.15%, *recall* 99.43%, dan *F1-Score* 99.29%, jauh melampaui model tanpa optimalisasi yang mendapat akurasi pada angka 81.88%. Model yang dioptimalisasi *hyperparameter* juga menunjukkan efisiensi waktu pelatihan yang lebih baik, hanya membutuhkan 9 *epoch* untuk mencapai konvergensi, dibandingkan dengan 50 *epoch* pada model tanpa optimalisasi *hyperparameter*. Penelitian ini membuktikan bahwa optimalisasi *hyperparameter* berperan penting dalam meningkatkan akurasi dan efisiensi model untuk mengidentifikasi gejala penyakit Monkeypox, serta membuka peluang untuk pengembangan alat bantu diagnostik berbasis kecerdasan buatan.

Kata kunci: Monkeypox, Deep Learning, EfficientNet-B1, Optimalisasi Hyperparameter.

IDENTIFICATION OF MONKEYPOX SYMPTOMS ON HUMAN SKIN USING EFFICIENTNET-B1 ARCHITECTURE WITH HYPERPARAMETER OPTIMIZATION

Abstract

Monkeypox is a zoonotic disease that requires early detection for proper treatment. This study designs a Monkeypox symptom identification model on human skin using EfficientNet-B1 architecture with hyperparameter optimization. The dataset used consists of 3,192 images divided into two classes, namely Monkeypox and Non-Monkeypox. The model design process involves the stages of problem identification, data collection, data pre-processing, model architecture design and hyperparameter implementation, model evaluation, and result analysis. Performance evaluation is done by comparing models that use hyperparameter optimization with models that are not optimized, using confusion matrix. The results show that the model with hyperparameter optimization achieves excellent performance with an accuracy value of 99.22%, precision 99.15%, recall 99.43%, and F1-Score 99.29%, far exceeding the model without optimization which gets accuracy at 81.88%. The hyperparameter optimized model also showed better training time efficiency, requiring only 9 epochs to reach convergence, compared to 50 epochs in the model without hyperparameter optimization. This research proves that hyperparameter optimization plays an important role in improving the accuracy and efficiency of the model to identify Monkeypox symptoms, and opens up opportunities for the development of artificial intelligence-based diagnostic tools.

Keywords: Monkeypox, Deep Learning, EfficientNet-B1, Hyperparameter Optimization

1. PENDAHULUAN

Monkeypox adalah penyakit *zoonosis* yang disebabkan oleh virus cacar monyet spesies dari genus *Orthopoxvirus* keluarga *Poxviridae* (Taruno et

al., 2023). Pertama kali ditemukan di antara koloni monyet yang dipelihara untuk penelitian pada tahun 1958. Kasus pertama pada manusia terjadi di Republik Demokratik Kongo pada tahun 1970 (Thornhill et al., 2022) dan mulai menyebar di

belahan Afrika Barat dan Afrika Tengah. Pada tahun 2022, terjadi peningkatan kasus Monkeypox yang signifikan di berbagai negara di luar Afrika, di mana penyakit ini sebelumnya bersifat endemik dan menimbulkan kekhawatiran akan potensi pandemi baru setelah pandemi COVID-19 (Kozlov, 2022). Gejala penyakit Monkeypox memiliki kemiripan dengan gejala cacar lainnya (Susanto, Nilogiri and Handayani, 2023). Kedua penyakit ini menyebabkan ruam pada kulit yang berubah menjadi lepuhan berisi cairan, yang kemudian berubah menjadi keropeng (Ng, 2022). Gejala awal dari penyakit ini serupa dengan demam, sakit kepala, dan mudah kelelahan. Cacar menular antar sesama manusia, sedangkan penularan penyakit Monkeypox dimulai dari hewan ke manusia, kemudian menyebar antara manusia. Deteksi dini gejala penyakit Monkeypox, terutama pada kulit, sangat penting agar dapat melakukan penanganan yang tepat dan pencegahan penyebaran. Namun, diagnosis visual menjadi tantangan karena adanya kemiripan gejala Monkeypox dengan penyakit kulit lainnya (Ogoina FMCP et al., 2020).

Dalam beberapa tahun terakhir, penelitian terkait deteksi penyakit kulit, termasuk gejala Monkeypox pada kulit manusia menggunakan teknologi kecerdasan buatan (AI) dan *deep learning* telah mengalami perkembangan yang signifikan. Penerapan *deep learning* dengan pemodelan klasifikasi gambar telah menunjukkan potensi besar dalam meningkatkan akurasi dan kecepatan diagnosis berbagai kondisi dermatologis. Contohnya, teknologi ini telah digunakan untuk menganalisis gambar dermatoskopi untuk mengklasifikasikan jenis-jenis tahi lalat, yang membantu deteksi dini melanoma (Gajera, Nayak and Zaveri, 2023). Kemampuannya dalam memproses dan menganalisis data visual dalam jumlah besar membuka peluang dalam diagnosis otomatis. Dibandingkan dengan metode konvensional, *deep learning* mampu mengidentifikasi pola yang lebih halus dan kompleks pada citra medis, seperti perbedaan tekstur dan warna yang sulit dideteksi oleh mata manusia, sehingga meningkatkan akurasi diagnosis. Potensi ini mendorong penelitian ini untuk mengembangkan model yang dapat mengklasifikasikan gambar medis secara otomatis, khususnya dalam mendiagnosis gejala penyakit Monkeypox.

Penelitian oleh Rohmatullah et al. menggunakan metode *knowledge discovery in database* (KDD) dengan algoritma *Support Vector Machine* (SVM) menggunakan 4 kernel yaitu linear, RBF, *sigmoid*, dan *polynomial* untuk mengklasifikasikan penyakit Monkeypox (Wijaya, Qur'ania and Anggraeni, 2024). Penelitian tersebut mencapai akurasi tertinggi di angka 75% dari kernel *polynomial*. Penelitian ini menggunakan dataset 1000 baris, yang kemungkinan besar menghadapi keterbatasan untuk menangkap kompleksitas penyakit Monkeypox. Keterbatasan ini dapat menyebabkan *overfitting*, di mana model bekerja baik

pada data pelatihan tetapi kurang akurat pada data baru. Hal ini ditunjukkan oleh akurasi pelatihan kernel RBF yang tinggi tetapi akurasi pengujianya rendah, yaitu 66%. Selain itu, penelitian ini tidak membahas bagaimana model SVM dapat diimplementasikan dalam pengaturan klinis. Penelitian terkait juga dilakukan oleh Fahrettin et al. membangun model untuk mendeteksi gejala Monkeypox secara akurat dan efisien. Metode *Multiple Nested Patch Division and Pretrained DenseNet201* dipilih sebagai dasar pengembangan model (Demir et al., 2024). Hasil akurasi klasifikasi dari model ini sebesar 91.87%, dengan nilai *recall* 91.49%, nilai *matric precision* 91.73%, dan nilai *f1-score* 91.60%. Meskipun demikian, penelitian ini memiliki keterbatasan. Dataset yang digunakan relatif kecil, terdiri dari 910 gambar yang diklasifikasikan menjadi 4 kategori yaitu *Monkeypox*, *Chickenpox*, *Smallpox*, dan *zoster zona*. Gambar dari penyakit Monkeypox sendiri hanya 217 gambar, sehingga kurang mampu untuk menangkap variabilitas dalam presentasi Monkeypox. Meskipun hasil akurasinya tinggi, model SVM yang digunakan masih sederhana untuk mengklasifikasikan gambar. Penggunaan metode klasifikasi yang lebih canggih atau melakukan pengoptimalan *hyperparameter* SVM akan memberi peluang untuk meningkatkan efektivitas dan akurasi model secara keseluruhan. Pendekatan inovatif dalam mendiagnosis gejala Monkeypox juga dapat dicapai melalui penggunaan jaringan saraf tiruan.

Penelitian yang memanfaatkan jaringan saraf tiruan, khususnya arsitektur *Convolution Neural Network* (CNN) telah banyak digunakan untuk mendeteksi penyakit Monkeypox. Penelitian oleh Murat Altun et al. membangun model untuk mendeteksi penyakit Monkeypox menggunakan *Convolutional Neural Network* dengan *Transfer Learning* (Altun et al., 2023). Berbagai model CNN diimplementasikan dalam penelitian ini, seperti *MobileNetV3-s*, *EfficientNetV2*, *ResNet50*, *Vgg19*, *DenseNet121*, dan model *Xception*. Dari proses yang dilakukan, model *MobileNetV3-s* mencapai akurasi tertinggi yaitu 96%. Meskipun model mencapai akurasi tinggi, penelitian ini mengamati bahwa *overfitting* terjadi ketika jumlah *epoch* pelatihan ditingkatkan. Untuk mengatasi masalah tersebut, *hyperparameter* model harus dioptimalkan lebih lanjut. Penelitian lain oleh Putu Desiana et al. menggunakan *SqueezeNet Feature Extraction* dan *Gradient Boosting* untuk menghasilkan model yang dapat mengklasifikasikan penyakit Monkeypox pada citra kulit (Desiana, Ayu and Pradipta, 2024). Model ini mencapai akurasi 94% dan metode *Gradient Boosting* terbukti dapat meningkatkan akurasi diagnostik untuk penyakit Monkeypox. Keterbatasan penelitian ini adalah tidak adanya analisis longitudinal untuk melacak kinerja model dalam jangka waktu yang lebih panjang.

Penelitian ini akan membangun model klasifikasi Monkeypox menggunakan arsitektur EfficientNet-B1, salah satu varian CNN. Optimalisasi *hyperparameter* akan diterapkan untuk meningkatkan performa model. *Convolutional Neural Network* (CNN) memanfaatkan operasi konvolusi untuk mengekstrak fitur dari gambar sebagai data input, yang membantu dalam mengenali pola dan membuat prediksi (Wang et al., 2022)(Oyededeji, Khan and Erkoyuncu, 2024). Fitur-fitur ini kemudian diklasifikasikan menggunakan jaringan saraf berlapis-lapis, yang membantu dalam mengenali pola dan membuat keputusan berdasarkan data yang diekstraksi (Pramanik et al., 2023). Metode CNN menunjukkan kemampuannya untuk mempertahankan akurasi dan ketahanan tinggi dalam tugas pengenalan gambar, sehingga banyak diterapkan dalam berbagai penelitian, terutama di bidang kesehatan.

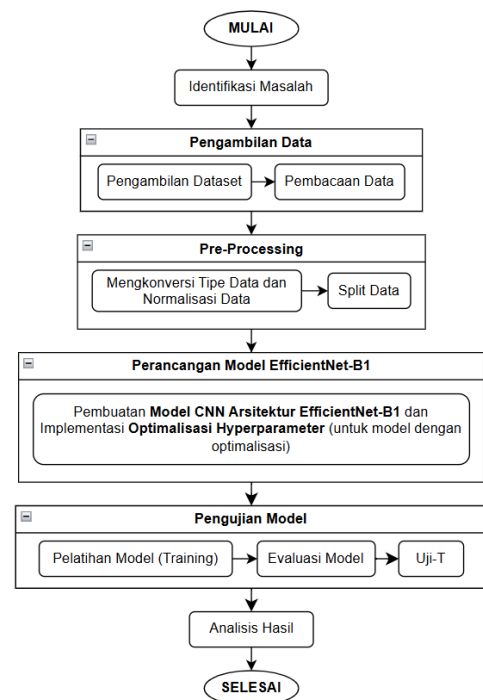
EfficientNet-B1 merupakan varian dari arsitektur CNN yang dirancang menggunakan teknik *Neural Architecture Search* (NAS) untuk mengoptimalkan *trade-off* antara kedalaman, lebar, dan resolusi jaringan. Hal tersebut mendukung keseimbangan optimal antara akurasi dan efisiensi komputasi dalam klasifikasi gambar (Mellian Ramadhan et al., 2023). Dengan teknik *compound scaling*-nya, EfficientNet-B1 memberikan fleksibilitas untuk menyesuaikan model dengan kebutuhan spesifik identifikasi gejala Monkeypox. Arsitektur ini berpotensi unggul dalam ekstraksi fitur dari gambar kulit pada berbagai skala yang penting untuk mengidentifikasi variasi gejala Monkeypox. Selain itu, EfficientNet-B1 juga menawarkan keseimbangan yang baik antara akurasi dan efisiensi komputasi. Keseimbangan ini merujuk pada kemampuan model untuk memberikan hasil prediksi yang akurat tanpa memerlukan sumber daya komputasi yang besar, seperti waktu pemrosesan yang lama atau penggunaan memori yang berlebihan. Menggunakan model yang terlalu kompleks atau memerlukan daya komputasi besar dapat menghambat efisiensi dan biaya, sedangkan model yang terlalu sederhana dapat mengorbankan akurasi. Penelitian sebelumnya telah menunjukkan bahwa menjaga keseimbangan ini adalah tantangan yang sering dihadapi, di mana algoritma yang sangat kompleks dapat mencapai akurasi tinggi tetapi dengan biaya efisiensi yang tinggi pula (Martha and Bororing, 2024). Oleh karena itu, sangat penting mengembangkan model yang memiliki keseimbangan antara akurasi dan efisiensi agar teknologi *deep learning* bisa diterapkan dengan baik di berbagai bidang. Meskipun demikian, optimalisasi *hyperparameter* tetap diperlukan untuk memaksimalkan kinerja model pada dataset spesifik. Proses ini melibatkan penyesuaian berbagai parameter seperti *learning rate*, *batch size*, dan jumlah *epoch*, yang semuanya berpengaruh signifikan terhadap akurasi klasifikasi.

Fokus utama penelitian ini adalah mengevaluasi efektivitas arsitektur EfficientNet-B1 dalam mengklasifikasikan gejala Monkeypox pada kulit manusia, dengan tingkat akurasi di atas 90%. Akurasi ini menjadi indikator kuat bahwa model memiliki kemampuan yang sangat baik dalam mengidentifikasi penyakit (Fahmi Idris et al., 2024). Selain itu, penelitian ini juga bertujuan mengidentifikasi parameter-parameter kunci yang mempengaruhi performa model serta membandingkan pengaruh optimalisasi *hyperparameter* terhadap akurasi dan efisiensi model. Dalam konteks medis, akurasi yang tinggi sangat krusial untuk memastikan diagnosis yang tepat dan pengobatan yang sesuai.

Harapan penelitian ini adalah dapat mengidentifikasi parameter-parameter yang berpengaruh signifikan terhadap performa model dalam mengidentifikasi gejala Monkeypox. Sehingga, model dapat dioptimalkan untuk mencapai tingkat akurasi, stabilitas, dan efisiensi yang tinggi. Dengan pemahaman mendalam terhadap parameter yang relevan, hasil klasifikasi akan menjadi lebih andal dan aplikatif dalam skenario nyata. Efektivitas model yang mencapai lebih dari 90% akan menunjukkan kemampuan model dalam menangkap pola kompleks pada data dan membuka peluang untuk penelitian lanjutan yang lebih mendalam terkait permasalahan yang diteliti.

2. METODE PENELITIAN

Alur perancangan model untuk identifikasi gejala penyakit Monkeypox pada kulit manusia dengan arsitektur EfficientNet-B1 ditampilkan pada Gambar 1.



Gambar 1. Alur Penelitian

2.1 Identifikasi Masalah

Tahap ini merupakan langkah awal penelitian, dimana peneliti mengidentifikasi dan merumuskan permasalahan yang akan diteliti. Ide yang diangkat merupakan fenomena yang sedang terjadi beberapa tahun terakhir, yaitu terkait penyakit Monkeypox seperti yang dijelaskan pada bab Pendahuluan. Setelah menentukan topik, peneliti mengumpulkan berbagai sumber literatur sebagai landasan untuk menemukan celah penelitian yang baru.

2.2 Pengambilan Data

Pada tahap ini, dilakukan pengambilan data yang relevan untuk menyelesaikan masalah yang telah diidentifikasi. Tahap ini bertujuan untuk mendapatkan dataset yang komprehensif, terdiri dari gambar-gambar kulit manusia yang bervariasi dalam hal ras, usia, jenis kelamin, dan tingkat keparahan gejala yang dialami.

2.2.1 Pengambilan Dataset

Penelitian ini menggunakan dataset publik berupa gambar, yaitu *Monkeypox Skin Lesion Dataset* yang diambil dari Kaggle (TensorKitty et al., 2023). Dataset ini dipublikasikan pada tahun 2023 oleh Tensorkitty *dkk*.

2.2.2 Pembacaan Data

Dataset yang digunakan dalam penelitian ini terdiri dari 3.192 gambar yang terbagi menjadi dua kelas, yaitu kelas Monkeypox (gambar dengan gejala penyakit Monkeypox) dan kelas *Others* (gambar tanpa gejala penyakit Monkeypox). Kelas **Monkeypox** berjumlah 1.428 gambar, sedangkan kelas **Others** terdiri dari 1.764 gambar. Setiap gambar dalam dataset ini dilabeli dengan “Monkeypox” atau “Non Monkeypox” dengan ukuran setiap gambar 224x224. Visualisasi gambar sampel dataset yang digunakan ditampilkan pada Gambar 2 berikut.



Gambar 2. Visualisasi Sampel Gambar Dataset

2.3 Pre-Processing

Pre-processing adalah tahap awal dalam pengolahan data yang bertujuan untuk mempersiapkan data mentah menjadi data yang berkualitas, konsisten, dan siap digunakan untuk

proses lebih lanjut. Pada penelitian ini, tahap *pre-processing* mencakup proses konversi tipe data dan pembagian data (*split data*) ke subset yang sesuai.

2.3.1 Mengkonversi Tipe Data dan Normalisasi Data

Sebagai persiapan untuk pemodelan, gambar-gambar yang telah dikumpulkan dikonversi ke dalam format float32. Konversi ini dilakukan untuk memastikan kompatibilitas dengan operasi numerik yang kompleks dalam *deep learning*, serta untuk memudahkan dalam perhitungan gradien dan optimasi model. Selanjutnya, dilakukan normalisasi data dengan membagi setiap nilai piksel dengan 255.0. Normalisasi ini bertujuan untuk menskalakan nilai piksel ke rentang 0-1, sehingga mempercepat proses pelatihan, meningkatkan akurasi model, dan mencegah masalah *vanishing gradient*. Dengan demikian, model dapat lebih efektif dalam mengekstrak fitur-fitur relevan dari gambar untuk mengidentifikasi gejala Monkeypox.

2.3.2 Split Data

Pada tahap ini, dataset dibagi menjadi tiga subset menggunakan fungsi *train test split* dengan komposisi 72% untuk data pelatihan, 8% untuk data validasi, dan 20% untuk data pengujian. Pembagian ini dilakukan untuk memastikan evaluasi model yang objektif, menghindari bias, serta mencegah *overfitting*. Pembagian dataset dilakukan sebagai berikut:

- 1) Data pelatihan digunakan untuk melatih model agar dapat mengenali pola dan hubungan dalam data untuk membuat prediksi.
- 2) Data validasi digunakan untuk memantau kinerja model selama proses pelatihan model.
- 3) Data pengujian digunakan untuk mengukur kemampuan generalisasi model secara objektif.

Tabel 1. Distribusi Data Setelah Pembagian Data

Data	Class	Jumlah
Data Pelatihan (72%)	Monkeypox	1027 gambar
	Non-Monkeypox	1270 gambar
Data Validasi (8%)	Monkeypox	286 gambar
	Non-Monkeypox	353 gambar
Data Pengujian (20%)	Monkeypox	115 gambar
	Non-Monkeypox	141 gambar

2.4 Perancangan Model EfficientNet-B1

Pada penelitian ini, model EfficientNet-B1 dipilih sebagai arsitektur *Convolutional Neural Network* yang akan digunakan untuk mengidentifikasi gejala penyakit Monkeypox pada kulit manusia. Arsitektur ini adalah salah satu varian dari EfficientNet yang dirancang oleh Google untuk

mencapai keseimbangan antara akurasi dan efisiensi komputasi. Model ini mengadopsi teknik *compound scaling*, sebuah metode yang menyeimbangkan tiga dimensi utama jaringan CNN, yaitu kedalaman (*depth*), lebar (*width*), dan resolusi (*resolution*) secara proporsional untuk meningkatkan efisiensi penggunaan parameter dan komputasi tanpa mengurangi akurasi. Berikut ini adalah Gambar arsitektur EfficientNet-B1 yang digunakan pada penelitian ini.

Layer (type)	Output Shape	Param #
efficientnetb1 (Functional)	(None, 1280)	6,575,239
flatten (Flatten)	(None, 1280)	0
dense (Dense)	(None, 1824)	1,311,744
dense_1 (Dense)	(None, 512)	524,800
dropout (Dropout)	(None, 512)	0
dense_2 (Dense)	(None, 2)	1,026

Gambar 3. Arsitektur EfficientNet-B1

Cara kerja EfficientNet-B1:

- 1) **Input:** Gambar input akan melewati lapisan stem untuk menghasilkan fitur awal.
- 2) **Inverted Bottleneck MBConv:** Fitur-fitur awal ini kemudian diolah melalui beberapa blok *inverted bottleneck* MBConv. Setiap blok akan mengekstrak fitur-fitur yang lebih kompleks dan abstrak.
- 3) **Global Average Pooling:** Output dari blok terakhir akan dirata-ratakan secara global untuk menghasilkan representasi fitur berdimensi rendah.
- 4) **Fully Connected Layer:** Representasi fitur ini kemudian diumpankan ke lapisan *fully connected* untuk menghasilkan skor kelas.
- 5) **Softmax:** Skor kelas ini kemudian dinormalisasikan menggunakan fungsi softmax untuk mendapatkan probabilitas kelas.

Hyperparameter adalah parameter yang nilainya ditentukan sebelum proses pelatihan dimulai dan dapat mempengaruhi kinerja model secara signifikan (Morales-Hernández, Van Nieuwenhuyse and Rojas Gonzalez, 2023). Penyetelan *hyperparameter* bertujuan untuk mendapatkan konfigurasi terbaik yang menghasilkan akurasi tinggi dan mengurangi resiko *overfitting*. Setelan *hyperparameter* pada penelitian ini mencakup beberapa aspek, mulai dari pengaturan arsitektur model (EfficientNet-B1 dengan modifikasi lapisan tambahan), fungsi aktivasi (*LeakyRelu*), *optimizer* (AdamW dengan *learning_rate* 0.0003 dan *weight_decay* 1e-5), fungsi *loss* (*sparse_categorical_crossentropy*), hingga strategi pelatihan seperti *batch_size*, jumlah *epoch*, dan penggunaan *callbacks* (*early stopping* berdasarkan nilai akurasi).

2.5 Pengujian Model

Pengujian model merupakan tahap krusial dalam pembuatan model karena bertujuan untuk menilai performa dan kemampuan model dalam menyelesaikan tugas tertentu. Pada penelitian ini, pengujian dilakukan untuk melatih dan mengevaluasi kinerja model dalam mengidentifikasi gejala penyakit Monkeypox pada kulit manusia.

2.5.1 Pelatihan Model (Training)

Tahap pelatihan model adalah proses utama dalam membangun kemampuan model untuk mengenali pola dan karakteristik data. Model dilatih menggunakan data pelatihan yang telah disiapkan melalui proses *split data* pada tahap sebelumnya. Selama pelatihan, model mempelajari hubungan antara fitur-fitur input dan label output melalui proses iteratif yang disebut *epoch*. Setiap proses *epoch* mencakup langkah-langkah berikut:

- 1) **Forward Propagation:** Model memproses data input (data pelatihan) melalui lapisan-lapisan arsitekturnya untuk menghasilkan output prediksi.
- 2) **Loss Calculation:** Perhitungan kesalahan (*loss*) dengan membandingkan output prediksi model dengan label sebenarnya menggunakan fungsi *loss*.
- 3) **Backward Propagation:** Model memperbaharui bobot (*weights*) dan biasanya untuk mengurangi nilai *loss*. Proses ini adalah mekanisme inti yang memastikan model dapat belajar dan memperbaiki diri secara bertahap berdasarkan umpan balik dari data, sehingga meningkatkan performa model selama proses pelatihan.
- 4) **Parameter Update:** Parameter/bobot model diperbarui untuk meminimalkan nilai *loss* pada iterasi berikutnya.

Proses ini akan dilakukan secara berulang hingga mencapai jumlah *epoch* yang telah ditentukan. Selama proses pelatihan, penggunaan data validasi penting untuk memantau apakah model mengalami *overfitting* atau *underfitting*.

2.5.2 Evaluasi Model (Evaluation)

Setelah model dilatih model akan diuji menggunakan data pengujian yang belum pernah dilihat oleh model sebelumnya. Proses evaluasi penting dilakukan agar dapat memahami seberapa baik model dapat melakukan generalisasi pada data baru, sehingga performa model pada data uji dapat diukur secara objektif. Pada penelitian ini, kinerja model akan diukur menggunakan metrik evaluasi yang dihitung berdasarkan *confusion matrix*, yaitu akurasi, *precision*, *recall*, dan *F1-Score*. *Confusion matrix* terdiri dari empat komponen utama, yaitu:

- 1) **True Positive (TP)**: Model dengan benar memprediksi kelas positif (kasus Monkeypox yang benar diidentifikasi sebagai Monkeypox).
- 2) **True Negative (TN)**: Model dengan benar memprediksi kelas negatif (kasus Monkeypox yang benar diidentifikasi sebagai Non-Monkeypox)
- 3) **False Positive (FP)**: Model salah memprediksi kelas positif (kasus yang seharusnya Non-Monkeypox tetapi diprediksi sebagai kasus Monkeypox)
- 4) **False Negative (FN)**: Model salah memprediksi kelas negatif (kasus yang seharusnya Monkeypox tetapi diprediksi sebagai kasus Non-Monkeypox)

Berikut adalah formula untuk menghitung model evaluasi dengan *confusion matrix*:

- 1) Akurasi: Proporsi prediksi yang benar terhadap jumlah total jumlah data pengujian.

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+FN+TN} \quad (1)$$

- 2) *Precision*: Mengukur ketepatan prediksi positif.

$$\text{Precision} = \frac{TP}{TP+FP} \quad (2)$$

- 3) *Recall*: Proporsi data positif yang benar-benar diprediksi positif.

$$\text{Recall} = \frac{TP}{TP+FN} \quad (3)$$

- 4) *F1-Score*: Keseimbangan antara precision dan recall.

$$\text{F1-Score} = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \quad (4)$$

2.5.3 Uji Statistik T-Test

Uji-T merupakan metode statistik inferensial yang digunakan untuk mengevaluasi apakah terdapat perbedaan yang signifikan secara statistik antara dua kondisi atau perlakuan (Puteri and Silvanie, 2020). Dalam konteks penelitian ini, uji-t digunakan untuk membandingkan performa model identifikasi gejala Monkeypox sebelum dan sesudah optimalisasi *hyperparameter*. Jenis uji yang digunakan adalah **Paired T-Test** (uji-t berpasangan) karena dua kondisi diuji pada data yang sama secara berulang.

Pengujian ini menggunakan fungsi `ttest_rel()` dari library `scipy.stats`, dengan tingkat signifikansi ditetapkan pada $\alpha = 0.05$. Hasil uji dianggap signifikan apabila nilai $p_value < 0.05$, yang menunjukkan bahwa perbedaan akurasi antara model tanpa dan dengan optimalisasi *hyperparameter* tidak

terjadi secara kebetulan. Dengan demikian, apabila hasil uji signifikan, maka dapat disimpulkan bahwa optimalisasi *hyperparameter* memberikan dampak terhadap peningkatan performa model identifikasi gejala Monkeypox. Rumus *Paired T-Test* adalah sebagai berikut.

$$t = \frac{\bar{d}}{s_d / \sqrt{n}} \quad (5)$$

di mana:

- \bar{d} = rata-rata dari selisih dua sampel
- s_d = standar deviasi dari selisih dua sampel
- n = jumlah percobaan

2.6 Analisis Hasil

Pada tahap ini, hasil dari evaluasi model akan dianalisis untuk melihat dan menentukan seberapa baik model dalam mendeteksi gejala Monkeypox. Jika kinerja model belum memuaskan, akan dilakukan perbaikan pada tahap-tahap sebelumnya, seperti mengubah arsitektur model atau menyesuaikan *hyperparameter*. Perbandingan antara model dengan dan tanpa optimalisasi *hyperparameter* dilakukan untuk menunjukkan pengaruh optimalisasi.

3. HASIL DAN PEMBAHASAN

Hasil Training Model EfficientNet-B1 Tanpa Optimalisasi *Hyperparameter*

Pada pelatihan model tanpa optimalisasi *hyperparameter*, proses *training* dirancang untuk dijalankan hingga 50 *epoch*. Jumlah ini dipilih untuk memastikan model memiliki cukup waktu untuk mempelajari dan mengadopsi pola dari data *training*. Pelatihan ini bertujuan untuk menetapkan performa awal (*baseline performance*) model yang akan menjadi acuan dalam mengevaluasi dampak penyeteraan *hyperparameter*. Hasil *training* model tanpa optimalisasi *hyperparameter* terlihat pada Tabel 2 berikut:

Tabel 2. Hasil Training Model Tanpa Optimalisasi *Hyperparameter*

Epoch	Accuracy	Loss	val accuracy	val loss
1	0.5102	0.9005	0.4603	0.7262
2	0.5446	0.8009	0.4372	0.7996
3	0.5819	0.7437	0.4372	0.7628
4	0.5894	0.7183	0.5397	0.7187
5	0.6120	0.6819	0.5586	0.7098
...
45	0.8531	0.3552	0.7971	0.4631
46	0.8436	0.3497	0.8013	0.4595
47	0.8534	0.3362	0.7992	0.4563
48	0.8617	0.3376	0.8013	0.4550
49	0.8591	0.3314	0.8075	0.4510
50	0.8764	0.3190	0.8096	0.4504

Model menunjukkan peningkatan akurasi secara konsisten mulai dari 51.02% pada *epoch* pertama hingga mencapai 87.64% pada *epoch* ke-50. Sementara itu, validasi akurasi meningkat dari 46.03% pada *epoch* pertama hingga 80.96% pada

epoch ke-50, yang menunjukkan model mampu mempelajari pola dari data *training* dengan cukup baik. Pola peningkatan validasi akurasi terlihat stabil hingga mencapai puncaknya pada *epoch*-38 tanpa adanya indikasi *overfitting* yang signifikan. Namun demikian, perbedaan konsistensi antara nilai akurasi dengan nilai validasi akhir mengindikasikan bahwa model masih memiliki keterbatasan dalam hal generalisasi. Hal ini menunjukkan bahwa, meskipun performa model pada data *training* sudah baik, kemampuan model untuk menetapkan pola yang dipelajari pada data validasi belum sepenuhnya optimal.

Hasil Training Model EfficientNet-B1 dengan Optimalisasi *Hyperparameter*

Hasil *training* model dengan optimalisasi *hyperparameter* terlihat pada Tabel 3 berikut:

Tabel 3. Hasil Training Model Dengan Optimalisasi *Hyperparameter*

Epoch	Accuracy	Loss	val accuracy	val loss
1	0.7513	0.4893	0.4492	0.7133
2	0.9306	0.1788	0.5547	0.7962
3	0.9597	0.1186	0.5859	1.7144
4	0.9783	0.0744	0.8125	0.5242
5	0.9801	0.0567	0.9023	0.4398
6	0.9795	0.0553	0.9648	0.1359
7	0.9750	0.0974	0.9961	0.0139
8	0.9926	0.0309	0.9883	0.0229
9	0.9908	0.0228	0.9922	0.0377

Model dengan optimalisasi *hyperparameter* menunjukkan peningkatan signifikan dalam akurasi, baik pada data *training* maupun data validasi, dibandingkan dengan model tanpa optimalisasi. Akurasi *training* meningkat dengan cepat sejak *epoch* pertama hingga ke-3, meskipun akurasi validasi pada tahap tersebut masih tertinggal. Namun, mulai dari *epoch* ke-4, model menunjukkan kemampuan generalisasi yang baik, ditandai dengan peningkatan akurasi validasi yang konsisten hingga mendekati akurasi *training*.

Pada tahap awal pelatihan, *loss* validasi mengalami lonjakan tajam, khususnya pada *epoch* ke-3 dengan nilai 1.7144. Hal ini mengindikasikan adanya kesenjangan antara data *training* dan data validasi. Namun, kesenjangan ini berhasil diminimalkan pada *epoch-epoch* berikutnya, hingga menjadi nilai *loss* akhir sebesar 0.0228. Nilai ini mencerminkan bahwa model telah mempelajari pola data secara optimal dan memiliki generalisasi yang sangat baik.

Kinerja terbaik dicapai pada *epoch* ke-9 dengan akurasi *training* sebesar 99.08% dan akurasi validasi sebesar 99.22% dari total 50 *epoch* yang ditentukan sebelumnya. Selain itu, model dengan optimalisasi *hyperparameter* menunjukkan efisiensi waktu pelatihan yang lebih baik, dengan durasi per *epoch* berkisar antara 19-23 detik dan konvergensi yang lebih cepat, yaitu hanya dalam 9 *epoch*.

Perbandingan Hasil Evaluasi Menggunakan Confusion Matrix

Tabel 4 berikut menyajikan perbandingan hasil evaluasi kedua model menggunakan *confusion matrix*.

Tabel 4. Perbandingan Hasil Confusion Matrix Kedua Model

Metrik	Model Tanpa Optimalisasi <i>Hyperparameter</i>	Model dengan Optimalisasi <i>Hyperparameter</i>
True Positive (TP)	238	351
True Negative (TN)	155	283
False Positive (FP)	56	3
False Negative (FN)	31	2

Hasil evaluasi menunjukkan bahwa optimalisasi *hyperparameter* secara signifikan meningkatkan kinerja model dalam mengklasifikasikan kasus Monkeypox. Pada model yang telah dioptimalisasi, jumlah *True Positive* (TP) meningkat dari 238 menjadi 351 kasus, mencerminkan kemampuan model yang lebih baik dalam mengidentifikasi kasus Monkeypox secara akurat. Sementara itu, jumlah *True Negative* (TN) juga meningkat dari 155 menjadi 283 kasus, menunjukkan perbaikan dalam deteksi kasus Non_Monkeypox dengan benar.

Penurunan kesalahan prediksi juga terlihat jelas, dengan jumlah *False Positive* (FP) berkurang drastis dari 56 menjadi hanya 3 kasus, menandakan kemampuan model dalam mengurangi kesalahan klasifikasi pada kasus Non-Monkeypox. Begitu pula, jumlah *False Negative* (FN) berkurang dari 31 menjadi 2 kasus, menunjukkan peningkatan signifikan dalam mendeteksi kasus Monkeypox yang sebelumnya terlewat.

Perbandingan Metrik Evaluasi

Tabel 5. Perbandingan Hasil Metrik Evaluasi Kedua Model

Model	Akurasi	Precision	Recall	F1-Score
Tanpa Optimalisasi	81.88%	80.95%	88.48%	84.54%
Dengan Optimalisasi	99.22%	99.15%	99.43%	99.29%

Tabel 5 menunjukkan perbandingan komprehensif metrik evaluasi antara model tanpa optimalisasi dan model dengan optimalisasi *hyperparameter*. Hasil analisis menunjukkan peningkatan yang signifikan pada semua metrik evaluasi setelah dilakukan optimalisasi.

Akurasi model mengalami peningkatan substansial dari 81.88% menjadi 99.22%,

menunjukkan perbaikan sebesar 17.34% dalam ketepatan klasifikasi secara keseluruhan. *Precision* model meningkat dari 80.95% menjadi 99.15%, mengindikasikan peningkatan sebesar 18.20% dalam kemampuan model untuk menghindari *false positive*.

Recall model juga menunjukkan peningkatan signifikan dari 88.48% menjadi 99.43%, yang menandakan adanya perbaikan sebesar 10.95% dalam kemampuan model untuk mendeteksi kasus positif yang sebenarnya. *F1-Score* yang merepresentasikan keseimbangan antara *precision* dan *recall*, meningkat dari 88.48% menjadi 99.29%, menunjukkan peningkatan sebesar 14.75%.

Peningkatan pada seluruh metrik evaluasi ini mengindikasikan bahwa proses optimalisasi *hyperparameter* berhasil meningkatkan performa model secara signifikan. Model yang dioptimalisasi menunjukkan kemampuan yang lebih baik dalam melakukan klasifikasi, dengan tingkat kesalahan yang jauh lebih rendah dibandingkan dengan model tanpa *hyperparameter*.

Perbandingan Hasil Model Terhadap Percobaan dengan Menggunakan Gambar Sebagai Input Prediksi

Pada penelitian ini, model yang digunakan untuk mengidentifikasi gejala Monkeypox pada kulit manusia menggunakan Arsitektur EfficientNet-B1 dari *Convolution Neural Network* (CNN). Proses perancangan model melibatkan beberapa tahapan, termasuk *pre-processing data*, perancangan model EfficientNet-B1, dan pengujian model. Model diuji menggunakan optimalisasi *hyperparameter*, sebagaimana dijelaskan pada bagian 2.4. Berikut adalah hasil perbandingan akurasi prediksi antara model tanpa optimalisasi *hyperparameter* dan model dengan optimalisasi *hyperparameter*.

Prediction: Non-Monkeypox
Accuracy: 55.59%



Gambar 4. Prediksi Menggunakan Model Tanpa Optimalisasi *Hyperparameter*

Pada gambar hasil prediksi dengan model tanpa optimalisasi *hyperparameter*, model memprediksi 'salah' bahwa gambar tidak menunjukkan gejala Monkeypox dengan akurasi 55,59%. Model tanpa optimalisasi *hyperparameter* cenderung memiliki performa yang kurang optimal karena parameter seperti *learning rate*, *batch size*, dan jumlah *epoch*

tidak disetel, sehingga model tidak dapat menangkap pola dari data secara efektif.

Prediction: Monkeypox
Accuracy: 99.66%



Gambar 5. Prediksi Menggunakan Model Dengan Optimalisasi *Hyperparameter*

Gambar 5 diatas menunjukkan bahwa model berhasil mengklasifikasikan gambar sebagai kasus Monkeypox dengan akurasi 99.66%. Dengan penyetelan *hyperparameter* seperti pada pembahasan sebelumnya, membuat model dapat belajar lebih efektif dan berhasil menangkap pola yang lebih kompleks dengan akurasi yang lebih tinggi. Selain itu, model dengan optimalisasi *hyperparameter* memiliki kemampuan generalisasi yang lebih baik, sehingga mampu membedakan gambar yang menunjukkan gejala Monkeypox dan yang tidak, dengan tingkat kesalahan yang lebih rendah.

Uji Statistik T-Test untuk Kedua Model

Nilai akurasi model pada lima kali percobaan tanpa dan dengan optimalisasi *hyperparameter* disajikan dalam Tabel 6 berikut.

Tabel 6. Hasil Akurasi Model Tanpa dan Dengan Optimalisasi *Hyperparameter* untuk Uji-T

Model	Nilai Akurasi Pada Percobaan				
	1	2	3	4	5
Tanpa <i>Hyperparameter</i>	0.7625	0.8125	0.7375	0.7971	0.7167
Dengan <i>Hyperparameter</i>	0.9969	0.9922	0.9750	0.9922	0.9953

Hasil perhitungan Paired T-Test diperoleh nilai statistik:

- **T-Statistic: -13.3215**
- **P-Value: 0.0002**

Hasil Uji-T menunjukkan perbedaan yang signifikan dalam akurasi model tanpa dan dengan optimalisasi *hyperparameter* ($t = -13,3215$, $p < 0,0002$). Hasil ini mengindikasikan bahwa optimalisasi *hyperparameter* secara statistik meningkatkan performa model dalam mengidentifikasi gejala Monkeypox pada kulit manusia.

4. KESIMPULAN DAN SARAN

Berdasarkan hasil analisis dan penelitian yang telah dilakukan, dapat disimpulkan bahwa performa model *deep learning* berbasis arsitektur EfficientNet-B1 menunjukkan kemampuan yang baik dalam mengidentifikasi gejala Monkeypox pada kulit manusia. Model dasar tanpa optimalisasi *hyperparameter* mencapai akurasi sebesar 81.88%, dengan nilai *precision* 80.95%, *recall* 88.48%, dan *F1-Score* 84.54%. Meskipun model ini mampu melakukan klasifikasi gejala Monkeypox, performanya belum optimal dalam aspek generalisasi terhadap data baru.

Optimalisasi *hyperparameter* terbukti meningkatkan performa model secara signifikan dengan akurasi yang meningkat menjadi 99.22%, *precision* 99.15%, *recall* 99.43%, dan *F1-Score* 99.29%. Selain peningkatan akurasi, optimalisasi juga mempercepat konvergensi dan mengurangi *overfitting*, sehingga model lebih andal dalam melakukan klasifikasi. Meskipun waktu pelatihan per *epoch* lebih bervariasi, peningkatan akurasi dan generalisasi yang dihasilkan jauh lebih penting. Secara keseluruhan, model dengan optimalisasi *hyperparameter* membutuhkan waktu yang lebih singkat untuk mencapai performa yang jauh lebih baik. Dengan demikian, optimalisasi *hyperparameter* berperan krusial dalam pengembangan model *deep learning* terutama dalam meningkatkan akurasi dan performa model untuk tugas klasifikasi medis, seperti dalam identifikasi gejala Monkeypox pada kulit manusia.

Hasil uji-t independen juga memperkuat hasil penelitian ini dengan nilai t-statistik sebesar -13,3215 dan nilai p (p-value) sebesar 0,0002. Karena nilai p lebih kecil dari tingkat signifikansi 0,05, maka dapat disimpulkan bahwa terdapat perbedaan yang signifikan secara statistik antara performa model tanpa dan dengan optimalisasi *hyperparameter*. Artinya, penerapan optimalisasi *hyperparameter* tidak hanya meningkatkan akurasi secara praktis, tetapi juga terbukti signifikan secara statistik dalam meningkatkan performa model dalam mengidentifikasi gejala Monkeypox pada kulit manusia.

Peneliti selanjutnya disarankan untuk memperkaya variasi dataset dengan menyertakan gambar gejala penyakit yang memiliki kemiripan dengan Monkeypox. Hal ini bertujuan untuk meningkatkan kemampuan model dalam membedakan gejala Monkeypox dari penyakit lain yang serupa, sehingga menghasilkan prediksi yang lebih akurat. Selain itu, perlu dilakukan eksplorasi lebih lanjut terhadap berbagai variasi arsitektur EfficientNet. Dengan demikian, dapat diperoleh perbandingan yang lebih komprehensif mengenai efektivitas masing-masing arsitektur dalam mengidentifikasi gambar gejala Monkeypox. Berdasarkan hal tersebut, diharapkan dapat ditemukan model yang tidak hanya memiliki akurasi

tinggi dalam mengidentifikasi gejala Monkeypox, tetapi juga memiliki kemampuan generalisasi yang baik terhadap data baru.

Hasil penelitian ini juga diharapkan dapat menjadi landasan bagi pengembangan lebih lanjut dalam menciptakan alat bantu diagnostik berbasis kecerdasan buatan, seperti aplikasi mobile, atau perangkat medis yang mampu mendukung proses diagnosis Monkeypox secara otomatis dan *real-time*.

DAFTAR PUSTAKA

- ALTUN, M., GÜRÜLER, H., ÖZKARACA, O., KHAN, F., KHAN, J., & LEE, Y. 2023. Monkeypox Detection Using CNN with Transfer Learning. *Sensors*, 23(4). <https://doi.org/10.3390/s23041783>
- DEMIR, F. B., BAYGIN, M., TUNCER, I., BARUA, P. D., DOGAN, S., TUNCER, T., OOI, C. P., CIACCIO, E. J., & ACHARYA, U. R. 2024. MNP DenseNet: Automated Monkeypox Detection Using Multiple Nested Patch Division and Pretrained DenseNet201. *Multimedia Tools and Applications*, 83(30), 75061–75083. <https://doi.org/10.1007/s11042-024-18416-4>
- DESIANA, P., AYU, W., & PRADIPTA, G. A. 2024. SqueezeNet Feature Extraction dan Gradient Boosting untuk Klasifikasi Penyakit Monkeypox pada Citra Kulit. *Jurnal Sistem Dan Informatika (JSI)*.
- FAHMI IDRIS, J., RAMADHANI, R., MALIK MUTOFFAR, M., Tinggi Teknologi Bandung Alamat, S., Soekarno Hatta No, J., Lega, K., Bojongloa Kidul, K., Bandung, K., & Barat, J. 2024. Klasifikasi Penyakit Kanker Paru Menggunakan Perbandingan Algoritma Machine Learning. In *JMA* (Vol. 2, Issue 2).
- GAJERA, H. K., NAYAK, D. R., & ZAVERI, M. A. 2023. A comprehensive analysis of dermoscopy images for melanoma detection via deep CNN features. *Biomedical Signal Processing and Control*, 79, 104186. <https://doi.org/10.1016/j.bspc.2022.104186>
- KOZLOV, M. 2022. Monkeypox goes global: why scientists are on alert. <https://doi.org/10.1038/D41586-022-01421-8>, 606(7912), 15–16.
- MARTHA, G., & BORORING, G. 2024. Pengembangan Algoritma Machine Learning Untuk Mendeteksi Anomali Dalam Jaringan Komputer. *Jurnal Review Pendidikan Dan Pengajaran*, 7(1364).
- MELLIAN RAMADHAN, I., SANJAYA, S., YANTO, F., & SYAFRIA, F. 2023. Image Classification of Beef and Pork Using Convolutional Neural Network Architecture EfficientNet-B1 1. *Indonesian Journal of Artificial Intelligence and Data Mining (IJAIMD)*, 6(1), 54–62. <https://doi.org/10.24014/ijaidm.v2i2>

- MORALES-HERNÁNDEZ, A., VAN NIEUWENHUYSE, I., & ROJAS GONZALEZ, S. 2023. A survey on multi-objective hyperparameter optimization algorithms for machine learning. In *Artificial Intelligence Review* (Vol. 56, Issue 8). Springer Netherlands. <https://doi.org/10.1007/s10462-022-10359-2>
- NG, J. (2022, September 1). *Perbedaan Cacar Monyet dan Cacar Air*. 01 Sep 2022. <https://aido.id/health-articles/perbedaan-cacarmonyet-dan-cacar-air/detail>
- OGOINA FMCP, D., IROEZINDU FWACP, M., IZIBEWULE JAMES MBBS, H., OLADOKUN FWACP, R., YINKA-OGUNLEYE MPH, A., WAKAMA MBBS, P., OTIKE-ODIBI FWACP, B., MUHAMMED USMAN MPH, L., OBAZEE FWACP, E., ARUNA FFPH, O., & IHEKWEAZU FFPH, C. 2020. Clinical course and outcome of human monkeypox in Nigeria. *Oxford University Press*. <https://doi.org/10.1093/cid/ciaa143/5734993>
- OYEDEJI, O. A., KHAN, S., & ERKOYUNCU, J. A. 2024. Application of CNN for multiple phase corrosion identification and region detection. *Applied Soft Computing*, 164. <https://doi.org/10.1016/j.asoc.2024.112008>
- PRAMANIK, R., BANERJEE, B., EFIMENKO, G., KAPLUN, D., & SARKAR, R. 2023. Monkeypox detection from skin lesion images using an amalgamation of CNN models aided with Beta function-based normalization scheme. *PLoS ONE*, 18(4 April). <https://doi.org/10.1371/journal.pone.0281815>
- PUTERI, K., & SILVANIE, A. 2020. Machine Learning untuk Model Prediksi Harga Sembako. *Jurnal Nasional Informatika*, 1(2), 82–94.
- SUSANTO, L. A., NILOGIRI, A., & HANDAYANI, L. 2023. Klasifikasi Citra Lesi Kulit Serupa Virus Monkeypox Menggunakan VGG-19 Convolutional Neural Network. *JUSTINDO (Jurnal Sistem Dan Teknologi Informasi Indonesia)*, 8(1), 1–9. <https://doi.org/10.32528/justindo.v8i1.168>
- TARUNO, P., NUGRAHA, G., DWIYANTSAPUTRA, R., & BIMANTORO, F. 2023. Monkeypox Classification based on Skin Images using CNN: EfficientNet-B0. *E3S Web of Conferences*, 465(02031). <https://doi.org/https://doi.org/10.1051/e3sconf/202346502031>
- TENSORKITTY, PAUL, J., AHMED TAZUDDIN, & PEANA TASNIM. 2023. *Monkeypox Skin Lesion Dataset*. Kaggle. https://www.kaggle.com/datasets/nafin59/monkeypox-skin-lesion-dataset?select=Monkeypox_Dataset_metadata.csv
- THORNHILL, J. P., BARKATI, S., WALMSLEY, S., ROCKSTROH, J., ANTINORI, A., HARRISON, L. B., PALICH, R., NORI, A., REEVES, I., HABIBI, M. S., APEA, V., BOESECKE, C., VANDEKERCKHOVE, L., YAKUBOVSKY, M., SENDAGORTA, E., BLANCO, J. L., FLORENCE, E., MOSCHESE, D., MALTEZ, F. M., ... ORKIN, C. M. 2022. Monkeypox Virus Infection in Humans across 16 Countries — April–June 2022. *New England Journal of Medicine*, 387(8), 679–691. <https://doi.org/10.1056/nejmoa2207323>
- WANG, Y., BAI, C., QIAN, X., LIU, W., ZHU, C., & GE, L. 2022. A DC Series Arc Fault Detection Method Based on a Lightweight Convolutional Neural Network Used in Photovoltaic System. *Energies*, 15(8). <https://doi.org/10.3390/en15082877>
- WIJAYA, R. S., QUR'ANIA, A., & ANGGRAENI, I. 2024. Klasifikasi Penyakit Cacar Monyet Menggunakan Support Vector Machine (SVM). *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, 4(4), 1253–1260. <https://doi.org/10.57152/malcom.v4i4.1417>