

Klasifikasi Spesies Jamur Beracun *Agaricus Xanthodermus* dan *Amanita Muscaria* Menggunakan Transfer Learning dengan Arsitektur MobileNetV2

Hildan Adisqi Ali Hasan¹, Risqy Siwi Pradini^{*2}, Mochammad Anshori³

^{1,2,3}Informatika, Fakultas Sains dan Teknologi, Institut Teknologi, Sains, dan Kesehatan RS.DR. Soepraoen Kesdam V/BRW, Indonesia

Email: ¹hildanalihasan@gmail.com, ²risqypradini@itsk-soepraoen.ac.id, ³moanshori@itsk-soepraoen.ac.id

Abstrak

Jamur memiliki peran penting dalam keanekaragaman hayati, namun beberapa spesies seperti *Agaricus xanthodermus* dan *Amanita muscaria* bersifat beracun dan dapat menyebabkan risiko kesehatan serius jika dikonsumsi. Identifikasi jamur beracun secara akurat menjadi tantangan karena kemiripan morfologinya dengan spesies non-beracun. Penelitian ini mengusulkan model klasifikasi jamur beracun menggunakan metode transfer learning dengan arsitektur MobileNetV2, yang dikenal efisien dalam memproses data visual. Dataset terdiri dari 632 gambar, masing-masing 304 gambar untuk *Agaricus xanthodermus* dan 328 gambar untuk *Amanita muscaria*, yang diperoleh dari platform Kaggle dan dibagi menjadi 80% data latih dan 20% data validasi. Augmentasi data seperti *rotation*, *shift*, *flipping*, dan *rescaling* diterapkan untuk meningkatkan generalisasi model. Eksperimen dilakukan dengan menguji pengaruh jumlah *epoch* terhadap performa model, menggunakan rentang 10 hingga 100 *epoch* dengan interval 10. Hasil menunjukkan bahwa akurasi model meningkat seiring bertambahnya jumlah *epoch*, dengan performa optimal pada *epoch* ke-60. Pada *epoch* ini, akurasi validasi mencapai 99.21% dengan nilai *loss* validasi terendah sebesar 0,0447, menunjukkan bahwa model mampu mengklasifikasikan kedua spesies jamur secara akurat dan efisien. Selain itu, tren akurasi dan *loss* pada data pelatihan menunjukkan bahwa model mampu belajar secara stabil dan tidak mengalami *overfitting*, bahkan ketika menggunakan dataset yang relatif kecil. Penelitian ini berkontribusi pada pengembangan metode klasifikasi jamur beracun yang lebih akurat dan efisien, yang memiliki implikasi penting dalam kesehatan masyarakat dan konservasi keanekaragaman hayati.

Kata kunci: *Agaricus Xanthodermus*, *Amanita Muscaria*, Klasifikasi, MobileNetV2, Transfer Learning.

Classification of Toxic Mushroom Species *Agaricus Xanthodermus* and *Amanita Muscaria* Using Transfer Learning with MobileNetV2 Architecture

Abstract

Mushrooms play an important role in biodiversity, but some species such as *Agaricus xanthodermus* and *Amanita muscaria* are toxic and can cause serious health risks if consumed. Accurate identification of toxic fungi is challenging due to their morphological similarity with non-poisonous species. This study proposes a toxic mushroom classification model using a transfer learning method with MobileNetV2 architecture, which is known to be efficient in processing visual data. The dataset consists of 632 images, 304 images each for *Agaricus xanthodermus* and 328 for *Amanita muscaria*, which were obtained from the Kaggle platform and divided into 80% training data and 20% validation data. Data augmentation such as *rotation*, *shift*, *flipping*, and *rescaling* were applied to improve the model's generalization. Experiments were conducted by testing the effect of the number of epochs on model performance, using a range of 10 to 100 epochs with an interval of 10. The results show that the model's accuracy increases as the number of epochs increases, with optimal performance at the 60th epoch. At this epoch, the validation accuracy reached 99.21% with the lowest validation loss value of 0.0447, indicating that the model could classify both fungal species accurately and efficiently. In addition, the accuracy and loss trends in the training data show that the model can learn stably and does not experience overfitting, even when using relatively small datasets. This research contributes to developing a more accurate and efficient method of toxic mold classification, which has important implications for public health and biodiversity conservation.

Keywords: *Agaricus Xanthodermus*, *Amanita Muscaria*, Classification, MobileNetV2, Transfer Learning.

1. PENDAHULUAN

Jamur merupakan aspek penting dari keanekaragaman hayati yang berkembang pesat di Indonesia. Jamur adalah kategori organisme unik yang tidak memiliki akar dan daun asli, membuatnya tidak mampu melakukan fotosintesis. Sifat jamur yang menarik terletak pada keragaman ukuran, bentuk, dan warnanya. Di seluruh dunia, terdapat sekitar 1.500.000 spesies jamur, dan sejauh ini, sekitar 74.000 spesies di antaranya telah berhasil diidentifikasi. Dari spesies yang telah diketahui, banyak di antaranya memberikan manfaat, seperti mendukung pertumbuhan ekonomi, meningkatkan praktik pertanian, dan memberikan kontribusi signifikan pada bidang kesehatan. Selain itu, jamur memiliki peran penting dalam kehidupan sehari-hari, tidak hanya sebagai pengurai utama dalam ekosistem, tetapi juga sebagai sumber pangan dan bahan obat yang esensial [1], [2].

Jamur merupakan sumber makanan yang kaya akan vitamin, mineral, dan nutrisi, rendah kalori, dan rendah lemak. Menurut beberapa ahli, jamur juga dapat membantu mencegah kanker payudara [2]. Meskipun jamur memiliki banyak manfaat, spesies jamur tertentu bersifat beracun yang dapat menyebabkan kematian. Jamur beracun yang merugikan ini diklasifikasikan ke dalam berbagai kelompok, seperti *Amatoxin/Amanatin (Cyclopeptida)*, *Gyromitrin*, *Orellanine*, *Asam Ibotenic*, *Muscimol*, *Psilocybin*, dan *Coprine* [3]. Setiap tahunnya di Indonesia, ada beberapa kasus kematian yang dikaitkan dengan keracunan jamur. Fenomena ini dapat dikaitkan dengan kurangnya pemahaman masyarakat Indonesia mengenai berbagai spesies jamur. Selama dekade sebelumnya, mulai dari 2010 hingga 2020, telah terdokumentasi 76 kasus keracunan akibat konsumsi jamur liar di Indonesia. Kasus ini mencakup total 550 individu yang terkena dampak, dan 9 diantaranya meninggal dunia [4]. Oleh karena itu, identifikasi spesies jamur beracun secara akurat memiliki signifikansi penting baik dalam bidang kesehatan masyarakat maupun ekologi [5].

Kurangnya pemahaman dan informasi yang akurat mengenai spesies jamur beracun menjadi masalah yang perlu diatasi. Masyarakat seringkali kesulitan membedakan antara spesies jamur yang aman untuk dikonsumsi dengan yang beracun. Hal ini dapat mengakibatkan kesalahan dalam pengumpulan dan konsumsi jamur liar, yang berujung pada kasus keracunan [6]. Selain itu, kurangnya edukasi mengenai pertolongan pertama dan penanganan yang tepat dalam kasus keracunan jamur juga menjadi masalah serius. Oleh karena itu, diperlukan upaya untuk meningkatkan kesadaran masyarakat mengenai bahaya jamur beracun, serta memberikan informasi yang akurat dan mudah diakses mengenai identifikasi, pencegahan, dan penanganan keracunan jamur.

Jamur beracun, seperti *Agaricus xanthodermus* [7] dan *Amanita muscaria* [8], dikenal memiliki potensi dampak serius terhadap kesehatan manusia jika dikonsumsi secara tidak sengaja. Efek toksisitas dari jamur beracun mencakup berbagai gejala yang mengancam jiwa, mulai dari gangguan pencernaan hingga kerusakan organ yang parah [9]. Dengan demikian, kemampuan untuk mengenali dan mengklasifikasikan spesies jamur beracun ini secara efisien menjadi hal yang sangat mendesak untuk dilakukan. Terdapat pendekatan tradisional yaitu observasi morfologi untuk klasifikasi spesies jamur. Pendekatan ini sering kali dianggap subjektif dan memerlukan keahlian khusus, sehingga tidak selalu dapat diandalkan dalam konteks yang membutuhkan respons cepat [10]. Dalam beberapa dekade terakhir, kemajuan teknologi kecerdasan buatan, khususnya dalam pengolahan citra digital dapat digunakan untuk mengatasi tantangan ini. Salah satunya adalah metode *transfer learning* yang memungkinkan pemanfaatan model yang telah dilatih sebelumnya. *Transfer learning* adalah sebuah metode dalam *machine learning*, khususnya *deep learning*, yang memanfaatkan pengetahuan yang telah diperoleh dari model yang dilatih sebelumnya pada tugas tertentu untuk diaplikasikan pada tugas baru yang memiliki keterkaitan [11]. Metode ini telah terbukti efektif dalam berbagai aplikasi klasifikasi visual, termasuk pengenalan objek biologis [12].

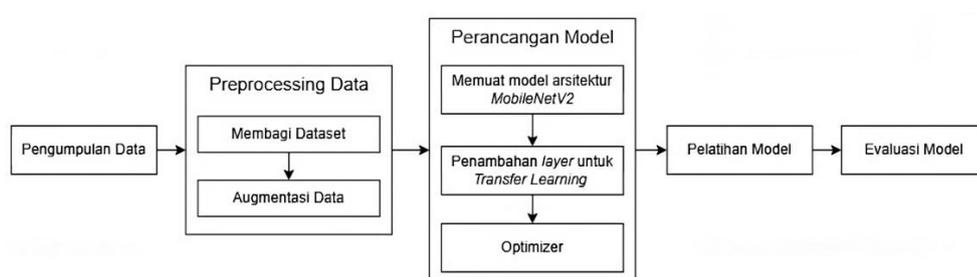
Seiring dengan perkembangan teknologi, metode *deep learning* telah muncul sebagai pendekatan yang sangat efektif dalam memecahkan masalah klasifikasi yang kompleks. Meskipun teknologi modern seperti *deep learning* memiliki potensi besar untuk meningkatkan akurasi dan efisiensi identifikasi, implementasi algoritma ini pada klasifikasi jamur sering kali menemui berbagai keterbatasan. Banyak penelitian sebelumnya hanya berfokus pada algoritma dasar atau pendekatan yang memerlukan dataset yang sangat besar dan sumber daya komputasi tinggi [13]. Oleh karena itu, diperlukan pendekatan yang tidak hanya mampu mengatasi kendala teknis ini, tetapi juga memberikan solusi yang dapat diandalkan dalam kondisi nyata, khususnya untuk klasifikasi spesies beracun seperti *Agaricus xanthodermus* dan *Amanita muscaria*.

Penelitian ini bertujuan untuk mengembangkan model klasifikasi berbasis *transfer learning* dengan arsitektur MobileNetV2 untuk mengidentifikasi spesies jamur beracun *Agaricus xanthodermus* dan *Amanita muscaria*. MobileNetV2 merupakan pilihan arsitektur yang unggul berkat efisiensi komputasi dan memorinya, yang dicapai melalui penggunaan *depthwise separable convolutions* dan *inverted residual blocks*. Arsitektur ini memungkinkan pemrosesan data visual dengan akurasi tinggi sekaligus menjaga model tetap ringan dan cepat. Hal ini menjadikannya pilihan yang tepat untuk tugas klasifikasi spesies jamur dengan dataset yang relatif terbatas [14]. Dengan menggunakan MobileNetV2, penelitian ini berfokus pada klasifikasi jamur beracun khususnya *Agaricus xanthodermus* dan *Amanita muscaria*. Model yang diusulkan akan dievaluasi berdasarkan akurasi dan efisiensinya dalam mengidentifikasi kedua spesies ini, yang dikenal memiliki potensi toksisitas tinggi dan sering kali salah

diidentifikasi. Selain itu, menurut kajian literatur menunjukkan bahwa kurangnya studi yang secara khusus memanfaatkan pendekatan *transfer learning* untuk mengidentifikasi spesies jamur dengan risiko toksisitas tinggi. Hal ini menciptakan celah penting dalam literatur, terutama dalam pengembangan metode otomatis yang tidak hanya akurat, tetapi juga ringan dan dapat diterapkan di lingkungan dengan sumber daya terbatas.

Oleh karena itu, penelitian ini menggunakan metode *transfer learning* dengan arsitektur MobileNetV2 untuk klasifikasi spesies jamur beracun. Pendekatan ini menawarkan keunggulan unik dibandingkan metode tradisional maupun algoritma *machine learning* konvensional, terutama dalam hal efisiensi komputasi dan kemampuan generalisasi pada dataset yang terbatas. Aspek kebaruan dari penelitian ini terletak pada penerapan *transfer learning* dengan arsitektur MobileNetV2 untuk klasifikasi spesies jamur yang sebelumnya kurang tereksplorasi, baik di literatur ilmiah maupun dalam aplikasi praktis. Selain itu, penelitian ini juga memberikan kontribusi penting dalam menyediakan model klasifikasi yang ringan namun tangguh, yang dapat digunakan untuk mendukung pengambilan keputusan cepat di lapangan, khususnya dalam konteks mitigasi risiko kesehatan akibat jamur beracun.

2. METODE PENELITIAN



Gambar 1. Diagram Alur Metode Penelitian

Berdasarkan Gambar 1, Metode penelitian yang digunakan untuk mengembangkan sistem klasifikasi jamur beracun dengan *transfer learning* melibatkan beberapa tahapan. Tahapan-tahapan klasifikasi tersebut meliputi pengumpulan data, *preprocessing* data yang terdiri dari pembagian dataset dan augmentasi data, perancangan model, dan pelatihan model. Lebih rincinya akan dijelaskan pada subbab dibawah.

2.1. Pengumpulan Data

Data yang digunakan dalam penelitian ini diperoleh dari platform Kaggle dengan judul dataset "*Mushroom dataset with a variety of species*". Dataset tersebut mencakup 94 spesies jamur yang beragam, menjadikannya sumber data yang kaya dan relevan untuk tugas klasifikasi berbasis computer vision. Dari keseluruhan data, penelitian ini memfokuskan pada dua spesies jamur beracun, yaitu *Agaricus xanthodermus* yang biasa disebut jamur pewarna kuning [7] dengan total 304 gambar dan *Amanita muscaria* lebih terkenal sebagai jamur agari lalat, dikenal karena topi merah ikoniknya yang dihiasi bintik-bintik putih [8] dengan total 328 gambar.



Gambar 2. (a) *Agaricus xanthodermus*, (b) *Amanita muscaria*

Merujuk pada Gambar 2 yaitu gambar jamur *Agaricus xanthodermus* dan *Amanita muscaria*, pemilihan kedua spesies ini dilakukan karena signifikansi biologisnya, serta untuk memberikan tantangan yang memadai dalam proses klasifikasi menggunakan metode *deep learning* yang sebelumnya belum pernah dilakukan. Data ini digunakan sebagai dasar untuk melatih dan menguji model klasifikasi, setelah melalui proses *preprocessing* untuk memastikan kualitas dan relevansi data terhadap tujuan penelitian.

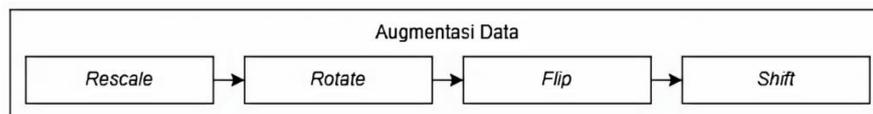
2.2. Preprocessing Data

Preprocessing data melibatkan dua langkah utama: pembagian data (*data splitting*) dan augmentasi data.

2.2.1. Pembagian Data

Pembagian data (*split data*) adalah proses membagi dataset menjadi beberapa bagian, yang terdiri dari data latih (*training data*) untuk melatih model dan data validasi (*validation data*) untuk menguji performa model selama pelatihan. Pada tahapan ini, dataset dibagi menjadi dua bagian berdasarkan proporsi total jumlah data, 80% dari seluruh dataset digunakan sebagai *data training* untuk melatih model, sementara 20% sisanya digunakan sebagai *data validation* untuk mengevaluasi performa model selama pelatihan. Pemilihan data dilakukan secara acak untuk memastikan representasi yang seimbang [15].

2.2.2. Augmentasi Data



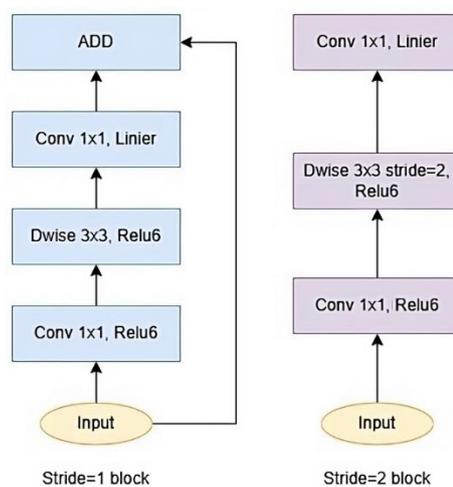
Gambar 3. Tahapan Augmentasi Data

Gambar 3 menggambarkan tahapan augmentasi data yang digunakan dalam penelitian ini. Proses augmentasi data menggunakan *ImageDataGenerator* dari *Keras*. Proses augmentasi dilakukan pada data pelatihan dengan menerapkan berbagai transformasi guna meningkatkan keragaman data dan kemampuan generalisasi model. Transformasi-transformasi tersebut merujuk pada Gambar 3 diatas dengan penjelasan sebagai berikut:

1. *Rescaling*: Semua piksel gambar diubah ke rentang 0 hingga 1. *Transformasi* ini memastikan data memiliki nilai yang konsisten untuk pelatihan model.
2. Rotasi: Gambar-gambar dilatih untuk memutar hingga 20 derajat secara acak untuk menangani variasi orientasi objek.
3. Pergeseran (*Shifting*): Gambar dipindahkan secara acak hingga 15% pada sumbu vertikal maupun horizontal untuk menangani variasi posisi objek.
4. *Flipping*: Gambar dibalik secara horizontal untuk mensimulasikan berbagai sudut pandang visual.

Sementara itu, data uji hanya di-*rescale* ke rentang 0 hingga 1 tanpa augmentasi tambahan untuk memastikan evaluasi model dilakukan pada data yang representatif terhadap kondisi asli. Teknik ini bertujuan untuk meningkatkan kinerja model pembelajaran mendalam dengan mengatasi *overfitting* dan meningkatkan generalisasi [16].

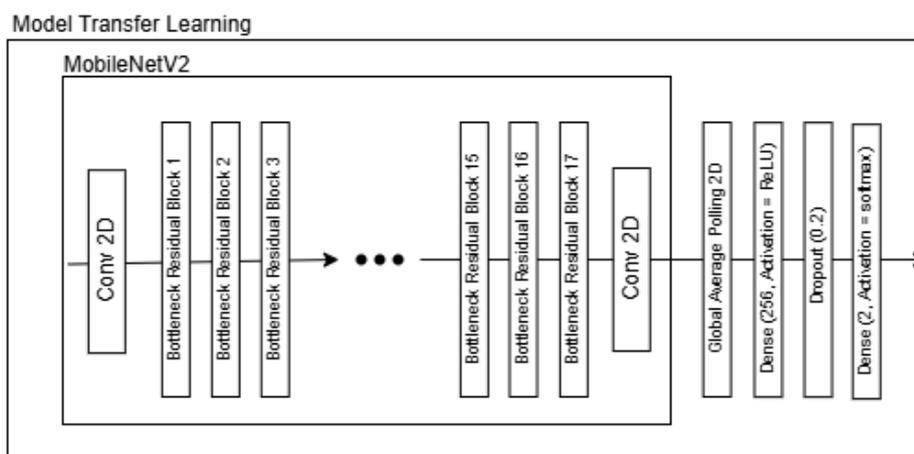
2.3. Perancangan Model



Gambar 4. Arsitektur dari MobileNetV2[17]

Tahap perancangan model dalam penelitian ini dimulai dengan memuat arsitektur MobileNetV2 sebagai model dasar untuk klasifikasi spesies jamur. MobileNetV2 merupakan salah satu arsitektur Convolutional Neural Network (CNN) yang dikembangkan oleh Google dengan fokus pada efisiensi dan kecepatan, terutama untuk perangkat dengan keterbatasan komputasi. MobileNetV2 dipilih karena keunggulannya sebagai model ringan yang efisien dalam memproses data gambar sambil tetap menghasilkan akurasi yang tinggi [17]. Arsitektur MobileNetV2 ditunjukkan pada Gambar 4 dengan memanfaatkan *inverted residual block*, yang terdiri dari dua jenis blok utama, blok dengan *stride 1* dan blok dengan *stride 2* [18]. Model ini kemudian diintegrasikan dengan lapisan klasifikasi tambahan yang disesuaikan untuk tugas dua kelas, yaitu *Agaricus xanthodermus* dan *Amanita muscaria*.

2.3.1. Penambahan Layer Untuk Transfer Learning



Gambar 5. Model transfer learning menggunakan MobileNetV2

Gambar 5 menyajikan arsitektur model *Transfer Learning* yang memanfaatkan MobileNetV2. Model ini diawali dengan lapisan *konvolusi 2D* (*Conv 2D*) yang berfungsi mengekstraksi fitur awal dari citra masukan. Selanjutnya, fitur-fitur ini diproses melalui serangkaian *Bottleneck Residual Block*, yang merupakan komponen kunci dari MobileNetV2. Blok-blok ini dirancang untuk meningkatkan efisiensi dan mengurangi kompleksitas komputasi dengan memanfaatkan konsep *depthwise separable convolution*. Diagram ini menunjukkan 17 blok *residual*, yang secara hierarkis mempelajari representasi fitur yang semakin abstrak dan kompleks. Setelah melalui blok-blok residual, fitur-fitur tersebut kembali diproses oleh lapisan *konvolusi 2D* (*Conv 2D*) sebelum dikirim ke bagian klasifikasi. Setelah fitur diekstraksi oleh MobileNetV2 akhirnya masuk ke tahap *Global Average Pooling 2D*. Tahap ini meratakan fitur-fitur spasial menjadi vektor fitur yang lebih ringkas. Selanjutnya, fitur yang telah dipadatkan tersebut diproses oleh lapisan *fully connected* dengan 256 neuron dan fungsi aktivasi *ReLU* untuk menangkap hubungan yang lebih kompleks dalam data. Untuk mengurangi risiko *overfitting*, diterapkan lapisan *Dropout* dengan rasio 0.2, yang secara acak menonaktifkan 20% neuron selama pelatihan agar model tidak terlalu bergantung pada fitur tertentu. Lapisan terakhir dalam model ini adalah lapisan *Dense* dengan dua neuron sesuai jumlah kelas yang diklasifikasikan, yaitu *Agaricus xanthodermus* dan *Amanita muscaria*. Fungsi aktivasi *softmax* digunakan pada lapisan ini untuk mengubah output menjadi probabilitas dari masing-masing kelas, sehingga memungkinkan model untuk melakukan klasifikasi dengan baik. Dengan memanfaatkan MobileNetV2 sebagai *backbone*, model ini dirancang agar efisien secara komputasi dengan tetap mempertahankan akurasi yang tinggi dalam mengklasifikasikan gambar jamur.

2.3.2. Optimizer

Pada tahap selanjutnya, model di-compile dengan menggunakan fungsi *loss categorical crossentropy*, yang dirancang untuk masalah klasifikasi multi-kelas tetapi juga dapat diadaptasi untuk tugas dua kelas dengan representasi *one-hot encoding*. Proses optimasi dilakukan menggunakan algoritma *Adam* (*Adaptive Moment Estimation*), yang merupakan salah satu *optimizer* adaptif terbaik untuk mempercepat konvergensi model. Parameter *learning rate* diatur sebesar $2e-4$ untuk mengontrol kecepatan pembaruan bobot pada setiap iterasi pelatihan [19]. Tahap *transfer learning* ini tidak hanya memastikan bahwa model MobileNetV2 dapat diadaptasi untuk dataset baru, tetapi juga meningkatkan efisiensi pelatihan dengan memanfaatkan arsitektur yang sudah

optimal. Dengan konfigurasi tersebut, model dapat belajar secara stabil dari dataset yang spesifik sambil meminimalkan risiko *overfitting* atau kesalahan pelatihan yang tidak terkendali [20].

2.4. Pelatihan Model

Proses pelatihan model dilakukan dengan menggunakan *data training* dan divalidasi menggunakan *data validation* pada setiap *epoch*. Jumlah siklus penuh ketika seluruh data training diproses oleh model disebut dengan *epoch* [20]. Dalam *transfer learning*, penggunaan *epoch* yang terlalu tinggi belum tentu memberikan peningkatan signifikan karena model sudah memiliki bobot awal dari pelatihan sebelumnya. Penelitian ini menguji jumlah *epoch* dalam rentang 10 hingga 100 dengan interval 10, untuk mengevaluasi performa model secara bertahap tanpa memperpanjang waktu pelatihan secara berlebihan. Pendekatan ini bertujuan menemukan jumlah *epoch* optimal yang seimbang antara akurasi dan efisiensi, sambil menghindari risiko *overfitting* pada dataset yang relatif kecil. Hasil pelatihan dibandingkan berdasarkan akurasi dan nilai *loss* pada *data validation* untuk menentukan konfigurasi terbaik.

2.5. Evaluasi Model

Evaluasi model dilakukan untuk memastikan kemampuan generalisasi pada data validasi. Metrik evaluasi meliputi akurasi dan *loss*:

$$\text{Akurasi} = \frac{\text{jumlah prediksi benar}}{\text{total data}} \quad (1)$$

$$\text{loss} = -\frac{1}{N} \sum_{i=1}^N \sum_{c=1}^C (p_{ic} \log(y_{ic})) \quad (2)$$

Akurasi digunakan untuk mengukur persentase prediksi yang benar dibandingkan total prediksi. Akurasi tinggi menunjukkan performa model yang baik dalam mengenali pola data. Formula dari akurasi ditunjukkan pada Persamaan (1). Sedangkan *loss* dipakai untuk mengukur sejauh mana prediksi model menyimpang dari nilai sebenarnya. *Loss* yang rendah mencerminkan kesalahan prediksi yang minimal. Formula dari *loss* ditunjukkan pada Persamaan (2). Dalam Formula tersebut, N adalah jumlah sampel dalam dataset, C adalah jumlah kelas, p_{ic} adalah probabilitas prediksi model untuk sampel ke- i pada kelas ke- c , dan y_{ic} adalah label sebenarnya dalam bentuk *one-hot encoding*. Fungsi ini bekerja dengan menghitung rata-rata negatif dari hasil perkalian antara probabilitas prediksi dan logaritma dari label sebenarnya. Jika model memberikan probabilitas tinggi pada kelas yang benar, nilai *loss* akan kecil, sedangkan jika model salah dalam prediksi, nilai *loss* akan besar. Oleh karena itu, optimasi model bertujuan untuk meminimalkan nilai fungsi ini agar meningkatkan akurasi prediksi klasifikasi [21].

3. HASIL DAN PEMBAHASAN

Bab ini menyajikan hasil dan pembahasan dari eksperimen yang telah dilakukan, merujuk pada metode penelitian yang telah dibahas sebelumnya. Tahapan-tahapan dalam penelitian ini meliputi pengumpulan data, *splitting data*, augmentasi data, dan pelatihan model *transfer learning* menggunakan arsitektur *MobileNetV2*. Eksperimen ini secara khusus menguji pengaruh nilai *epoch* terhadap performa model.

3.1. Pengumpulan dan Preprocessing Dataset

Data yang digunakan dalam penelitian ini diperoleh dari platform Kaggle dengan judul dataset "*Mushroom dataset with a variety of species*". Pembagian dataset ini bertujuan untuk memastikan bahwa model dilatih dengan data yang cukup representatif sekaligus dievaluasi kinerjanya pada data yang belum pernah dilihat sebelumnya. Proporsi 80% untuk data latih dan 20% untuk data validasi umum digunakan dalam penelitian *machine learning* untuk mencapai keseimbangan antara jumlah data yang cukup untuk pelatihan dan validasi yang memadai.

Dataset ini terdiri dari dua kelas, yaitu *Agaricus xanthodermus* dan *Amanita muscaria*, dengan total 632 data. Rincian sebaran data untuk tiap kelas adalah 304 data untuk kelas *Agaricus xanthodermus* dan 328 data untuk kelas *Amanita muscaria*. Data kemudian dibagi (*splitting*) dengan rasio 80% sebagai *data training* dan 20% sebagai *data validation* sebagaimana ditunjukkan pada Tabel 1.

Tabel 1. Pembagian Dataset

Kelas	Data Training	Data Validation	Total
Agaricus xanthodermus	243	61	304
Amanita muscaria	262	66	328
Total	505	127	632
Presentase	80%	20%	100%

Berdasarkan pada Tabel 1, Dataset ini terdiri dari dua kelas, yaitu *Agaricus xanthodermus* dan *Amanita muscaria*, dengan total 632 data. Rincian sebaran data untuk tiap kelas adalah 304 data untuk kelas *Agaricus xanthodermus* dan 328 data untuk kelas *Amanita muscaria*. Data kemudian dibagi (*splitting*) dengan rasio 80% sebagai *data training* dan 20% sebagai *data validation*.

Sebelum pelatihan model, data gambar dilakukan augmentasi menggunakan teknik-teknik seperti *rescale*, rotasi, *flip*, dan pergeseran (*shift*). *Rescale* dilakukan untuk mengubah nilai *piksel* gambar ke dalam rentang 0 hingga 1. Rotasi, *flip*, dan pergeseran bertujuan untuk meningkatkan variasi data pelatihan, sehingga model menjadi lebih *robust* dan mampu menggeneralisasi dengan baik pada data baru. Augmentasi data merupakan langkah penting dalam *deep learning*, terutama ketika jumlah data yang tersedia terbatas. Dengan memperbanyak variasi data pelatihan, model akan terhindari dari *overfitting* dan mampu belajar fitur-fitur yang lebih general dari objek yang dikenal

3.2. Pelatihan Model dan Pengujian Epoch

Model *transfer learning* dengan arsitektur MobileNetV2 dilatih menggunakan data yang telah di augmentasi. Eksperimen utama dalam penelitian ini adalah pengujian nilai *epoch* pada rentang dan interval tertentu. *Epoch* yang diuji berkisar antara 10 hingga 100 dengan interval 10.

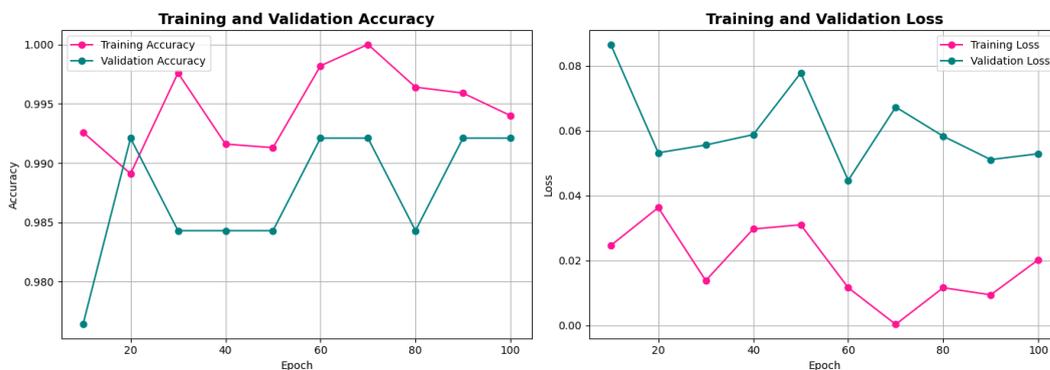
Pengujian *epoch* ini bertujuan untuk menemukan jumlah epoch optimal yang memberikan kinerja terbaik pada model. Setiap *epoch* merepresentasikan satu siklus pelatihan melalui seluruh dataset pelatihan. Semakin banyak *epoch*, model diharapkan belajar lebih lama dan lebih mendalam kemampuannya dalam mengenali pola. Namun, terlalu banyak *epoch* juga dapat menyebabkan *overfitting*.

Tabel 2. Perbandingan Akurasi dan Loss Terhadap Epoch

Epoch	Training Accuracy	Training Loss	Validation Accuracy	Validation Loss
10	0.9926	0.0246	0.9764	0.0866
20	0.9891	0.0363	0.9921	0.0532
30	0.9976	0.0138	0.9843	0.0556
40	0.9916	0.0297	0.9843	0.0588
50	0.9913	0.0310	0.9843	0.0778
60	0.9982	0.0116	0.9921	0.0447
70	1	0.0003	0.9921	0.0673
80	0.9964	0.0116	0.9843	0.0583
90	0.9959	0.0094	0.9921	0.0511
100	0.9940	0.0202	0.9921	0.0529

Tabel 2 di atas menyajikan tabel yang berisi perbandingan metrik kinerja antara *Training Accuracy*, *Training Loss*, *Validation Accuracy*, dan *Validation Loss* pada berbagai *epoch*. Data ini diperoleh dari hasil pelatihan model *deep learning* untuk klasifikasi suatu dataset. Pada epoch 10, terlihat bahwa model mencapai *Training Accuracy* sebesar 0.9926 dan *Validation Accuracy* sebesar 0.9764. Meskipun demikian, nilai *Training Loss* dan *Validation Loss* pada *epoch* ini relatif tinggi, yaitu 0.0246 dan 0.0866. Seiring dengan bertambahnya *epoch*, terjadi peningkatan signifikan pada *Training Accuracy* dan *Validation Accuracy*, sementara *Training Loss* dan *Validation Loss* mengalami penurunan. Peningkatan kinerja ini mencapai puncaknya pada *epoch* 60, di mana model mencatatkan *Training Accuracy* sebesar 0.9982 dan *Validation Accuracy* sebesar 0.9921, dengan *Training Loss* dan *Validation Loss* masing-masing sebesar 0.0116 dan 0.0447. Namun, setelah *epoch* 60, peningkatan kinerja model tidak lagi signifikan, bahkan cenderung fluktuatif. Pada *epoch* 70, *Training Accuracy* mencapai 1, tetapi *Validation Loss* juga meningkat menjadi 0.0673. Hal ini mengindikasikan bahwa model mulai mengalami *overfitting* setelah melewati *epoch* 60. Oleh karena itu, dapat disimpulkan bahwa *epoch* 60 merupakan titik optimal dalam pelatihan model ini, di mana model mencapai kinerja terbaiknya tanpa mengalami *overfitting*. Hasil menunjukkan bahwa MobileNetV2, dengan pendekatan transfer learning, mampu mencapai akurasi tinggi meskipun menggunakan dataset terbatas. Hal ini konsisten dengan penelitian sebelumnya yang menunjukkan bahwa transfer learning dapat meningkatkan akurasi pada tugas klasifikasi dengan dataset kecil [13], [17]. Model

menunjukkan kemampuan generalisasi yang baik pada data validasi, sebagaimana terlihat dari akurasi validasi yang stabil di atas 99%.



Gambar 6. Tren Akurasi dan Loss

Gambar 6. menunjukkan tren akurasi dan loss dari model selama proses pelatihan. Akurasi pelatihan meningkat hingga mencapai 100% pada epoch ke-70, sedangkan akurasi validasi stabil di kisaran 98–99%, dengan nilai tertinggi 99,21% pada epoch ke-60 dan 70. Grafik loss menunjukkan penurunan signifikan pada training loss, sedangkan validation loss terendah juga terjadi pada epoch ke-60, yaitu sebesar 0,0447. Hal ini menunjukkan bahwa model mampu belajar secara efektif tanpa overfitting, dan performa terbaik dicapai pada epoch ke-60.

Hasil menunjukkan bahwa MobileNetV2, dengan pendekatan transfer learning, mampu mencapai akurasi tinggi meskipun menggunakan dataset terbatas. Hal ini konsisten dengan penelitian sebelumnya yang menunjukkan bahwa transfer learning dapat meningkatkan akurasi pada tugas klasifikasi dengan dataset kecil [13], [17]. Model menunjukkan kemampuan generalisasi yang baik pada data validasi, sebagaimana terlihat dari akurasi validasi yang stabil di atas 99%.

4. KESIMPULAN

Penelitian ini berfokus pada klasifikasi dua spesies jamur beracun, yaitu *Agaricus xanthodermus* dan *Amanita muscaria*, yang dikenal memiliki kemiripan morfologi dengan spesies non-beracun. Metode yang digunakan adalah *transfer learning* dengan arsitektur MobileNetV2. Tahapan penelitian meliputi preprocessing data dengan augmentasi, seperti rotasi, *shift*, *flipping*, dan *resizing*, untuk meningkatkan keanekaragaman data, serta melatih model menggunakan pendekatan *transfer learning*. Eksperimen dilakukan dengan menguji pengaruh jumlah *epoch* terhadap performa model, menggunakan rentang *epoch* 10 hingga 100 dengan interval 10. Hasil penelitian menunjukkan bahwa model mencapai performa optimal pada *epoch* ke-60, dengan akurasi validasi tertinggi sebesar 99.21% dan nilai *loss* validasi terendah 0,0447. Keunggulan utama dari penggunaan MobileNetV2 terletak pada efisiensi komputasi dan kemampuannya untuk menghasilkan akurasi tinggi meskipun dengan dataset yang relatif kecil. Untuk penelitian selanjutnya, disarankan untuk mengeksplorasi penggunaan dataset yang lebih besar dan lebih bervariasi guna meningkatkan kemampuan generalisasi model. Selain itu, pengujian terhadap arsitektur model lain juga dapat dilakukan untuk membandingkan keunggulan dan kekurangan MobileNetV2 dalam klasifikasi spesies jamur beracun.

DAFTAR PUSTAKA

- [1] O. N. Putri, "Implementasi Metode CNN Dalam Klasifikasi Gambar Jamur Pada Analisis Image Processing (Studi Kasus: Gambar Jamur Dengan Genus *Agaricus* Dan *Amanita*)," *Skripsi*, pp. 1–80, 2020, [Online]. Available: <http://dspace.uui.ac.id/123456789/23677>
- [2] U. S. Rahmadhani and N. L. Marpaung, "Klasifikasi Jamur Berdasarkan Genus Dengan Menggunakan Metode CNN," *J. Inform. J. Pengemb. IT*, vol. 8, no. 2, pp. 169–173, 2023, doi: 10.30591/jpit.v8i2.5229.
- [3] I. P. Ihsan and F. Yusuf, "Analisis Jamur Beracun Berdasarkan Ciri Menggunakan Algoritma AdaBoost," in *Prosiding Konferensi Ilmu Komputer Nasional 2013*, 2013, pp. 1–6. [Online]. Available: https://www.researchgate.net/profile/Indah-Ihsan/publication/332440445_jurnal_KONIK_2013_Indah_Purwitasari_Ihsan/links/5cb5773692851c8d22ee1d06/jurnal-KONIK-2013-Indah-Purwitasari-Ihsan.pdf
- [4] I. P. Putra, "Kasus keracunan *Inocybe* sp. di Indonesia," in *Prosiding Seminar Nasional Biologi di Era Pandemi COVID-19*, Prosiding Seminar Nasional Biologi, 2020, pp. 148–153. doi: 2450

- <https://doi.org/10.24252/psb.v6i1.15727>.
- [5] Z. Salem-bango, T. K. Price, J. L. Chan, and O. B. G. andShangxin Y. Sukantha Chandrasekaran, "Fungal Whole-Genome Sequencing for Species Identification : From Test Development to Clinical Utilization," *Journal of Fungi*, vol. 9, no. 2, 2023, doi: <https://doi.org/10.3390/jof9020183>.
- [6] I. S. Kasus, "Identifikasi Jamur Beracun *Clitocybe sp . di Gresik ,*" pp. 119–124, [Online]. Available: <https://doi.org/10.22435/mpk.v3i1i2.4352>
- [7] M. Gill and Richard J. Strauch, "Constituents of *Agaricus xanthodermus* Genevier: The First Naturally Endogenous Azo Compound and Toxic Phenolic Metabolites," *Nature*, vol. 39, no. 11–12, p. 977, 1984, doi: <https://doi.org/10.1515/znc-1984-11-1203>.
- [8] D. Michelot and L. M. Melendez-Howell, "Amanita muscaria: Chemistry, biology, toxicology, and ethnomycology," *Mycol. Res.*, vol. 107, no. 2, pp. 131–146, 2003, doi: 10.1017/S0953756203007305.
- [9] A. D. L. Lima, R. C. Fortes, M. R. C. G. Novaes, and S. Percário, "Poisonous mushrooms ; a review of the most common intoxications," *Nutr. Hosp.*, vol. 27, no. 2, pp. 402–408, 2012, doi: 10.3305/nh.2012.27.2.5328.
- [10] P. W. Trail, "Forensic Science International : Animals and Environments Morphological analysis : A powerful tool in wildlife forensic biology," *Forensic Sci. Int. Anim. Environ.*, vol. 1, no. July, p. 100025, 2021, doi: 10.1016/j.fsiae.2021.100025.
- [11] D. Vu, M. Groenewald, and G. Verkley, "convolutional neural networks improve fungal classification," *Sci. Rep.*, pp. 1–12, 2020, doi: <https://doi.org/10.1038/s41598-020-69245-y>.
- [12] J. Knapheide, B. Stabernack, and M. Kuhnke, "A High Throughput MobileNetV2 FPGA implementation based on a Flexible Architecture for Depthwise Separable Convolution," in *2020 30th International Conference on Field-Programmable Logic and Applications (FPL)*, IEEE (Intitute of Electrical and Electronics Engineers), 2020, pp. 277–283. doi: 10.1109/FPL50879.2020.00053.
- [13] S. A. Munoz, J. Park, C. M. Stewart, A. M. Martin, and J. D. Hedengren, "Deep Transfer Learning for Approximate Model Predictive Control," *Processes*, vol. 11, no. 1, 2023, doi: 10.3390/pr11010197.
- [14] N. Pratama, Muhaza Liebenlito, and Yanne Irene, "PERBANDINGAN MODEL KLASIFIKASI TRANSFER LEARNING CONVOLUTIONAL NEURAL NETWORKTUMOR OTAKMENGUNAKAN CITRA MAGNETIC RESONANCE IMAGING," *J. Sehat Indones.*, vol. 6, no. 1, 2024, doi: <https://doi.org/10.59141/jsi.v6i01.81>.
- [15] V. R. Joseph, A. Vakayil, V. R. Joseph, and A. Vakayil, "SPLIT : An Optimal Method for Data Splitting SPLIT : An Optimal Method for Data Splitting ABSTRACT," *Technometrics*, vol. 64, no. 2, pp. 1–23, 2022, doi: 10.1080/00401706.2021.1921037.
- [16] C. Shorten and T. M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," *J. Big Data*, vol. 6, no. 60, 2019, doi: 10.1186/s40537-019-0197-0.
- [17] K. Dong, C. Zhou, Y. Ruan, and Y. Li, "MobileNetV2 Model for Image Classification," in *Proceedings - 2020 2nd International Conference on Information Technology and Computer Application, ITCA 2020*, 2020, pp. 476–480. doi: 10.1109/ITCA52113.2020.00106.
- [18] D. Hussain, M. Ismail, I. Hussain, R. Alroobaea, S. Hussain, and S. S. Ullah, "Face Mask Detection Using Deep Convolutional Neural Network and MobileNetV2-Based Transfer Learning," *Wirel. Commun. Mob. Comput.*, vol. 2022, 2022, doi: 10.1155/2022/1536318.
- [19] X. Jiang, B. Hu, S. Chandra Satapathy, S. H. Wang, and Y. D. Zhang, "Fingerspelling Identification for Chinese Sign Language via AlexNet-Based Transfer Learning and Adam Optimizer," *Sci. Program.*, vol. 2020, 2020, doi: 10.1155/2020/3291426.
- [20] S. Afaq and S. Rao, "Significance Of Epochs On Training A Neural Network," *Int. J. Sci. Technol. Res.*, vol. 9, no. 06, pp. 1–4, 2020, [Online]. Available: www.ijstr.org
- [21] A. Rusiecki, "Trimmed categorical cross-entropy for deep learning with label noise," *Electron. Lett.*, vol. 55, no. 6, pp. 319–320, 2019, doi: 10.1049/el.2018.7980.