



Klasifikasi Kualitas Biji Kedelai Menggunakan *Transfer Learning* *Convolutional Neural Network* Dan SMOTE

Amanda Prawita Ningrum¹, Sri Winarno², Vincentius Praskatama³

^{1,2,3} Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro, Semarang, Indonesia

¹111202113646@mhs.dinus.ac.id, ²sri.winarno@dsn.dinus.ac.id, ³111202113456@mhs.dinus.ac.id

Abstract

Soybeans are an important commodity in the food and feed industry, where they can be used to produce soy milk or other processed products. However, low-quality seeds can reduce the quality of the processed products and increase production costs. To address this issue, a soybean seed quality classification system was developed using the method of Transfer Learning with CNN and SMOTE. This method leverages the ability of neural networks for extracting of the visual features and handle for data imbalance between classes. The study shows that the CNN model achieved value accuracy is 91.09%, while the combination of CNN and SMOTE get the accuracy of 89.92%. Additionally, the MobileNetV2 model reached an accuracy of 91.11%, which further improved to 92.42% after applying SMOTE. These results demonstrate that the use of Transfer Learning and SMOTE significantly enhances accuracy in soybean seed quality classification, resulting in a more effective.

Keywords: Soybean Seed, CNN, Mobile Net, SMOTE, Transfer Learning

Abstrak

Kedelai merupakan komoditas penting dalam industri pangan dan pakan ternak, di mana kedelai bisa diolah menjadi susu kedelai atau produk olahan lainnya. Namun, biji kedelai berkualitas rendah dapat menurunkan kualitas produk olahan dan meningkatkan biaya produksi. Untuk mengatasi masalah ini, sistem klasifikasi kualitas biji kedelai dikembangkan menggunakan metode *transfer learning* dengan CNN dan SMOTE. Metode ini memanfaatkan kemampuan jaringan saraf tiruan untuk mengekstrak fitur visual serta mengatasi ketidakseimbangan data antar kelas. Penelitian menunjukkan bahwa model CNN mencapai akurasi sebesar 91,09%, sementara kombinasi CNN dan SMOTE memperoleh akurasi sebesar 89,92%. Selain itu, model *MobileNetV2* berhasil mencapai akurasi sebesar 91,11%, yang kemudian meningkat menjadi 92,42% setelah penerapan SMOTE. Hasil ini menunjukkan bahwa penggunaan *transfer learning* dan SMOTE secara signifikan meningkatkan akurasi dalam klasifikasi kualitas biji kedelai, sehingga memberikan hasil yang lebih efektif.

Kata kunci: Biji Kedelai, CNN, *Mobile Net*, SMOTE, *Transfer Learning*

1. Pendahuluan

Kedelai merupakan salah satu komoditas pertanian utama yang memiliki nilai ekonomi tinggi dan banyak digunakan dalam industri pangan dan pakan ternak [1]. Kaya akan protein, serat, dan asam lemak esensial, kedelai menjadi bahan baku utama untuk berbagai produk olahan seperti tempe, tahu, minyak kedelai, dan susu kedelai [2]. Selain itu, kedelai juga berperan sebagai bahan baku pakan ternak serta sumber protein nabati dalam produk makanan sehat [3]. Kualitas biji kedelai sangat menentukan keberhasilan produksi, karena biji berkualitas tinggi umumnya memiliki bentuk seragam, bebas dari kerusakan fisik, serta mengandung nutrisi yang optimal [4]. Namun, jika kualitas biji kedelai rendah, kerugian dapat terjadi di berbagai tahap. Dari perspektif petani, mereka menghadapi penurunan harga jual karena kualitas yang buruk tidak memenuhi standar pasar atau industri [5].

Selain itu, hasil panen yang tidak optimal menyebabkan biaya produksi lebih tinggi untuk menangani kerusakan fisik atau ketidakteraturan ukuran biji, seperti biaya tambahan untuk manajemen dan penyortiran lebih lanjut [6]. Penurunan harga jual ini berdampak langsung pada pendapatan petani, yang berkurang meskipun upaya dan biaya produksi tinggi. Dari perspektif penjual dan distributor, produk berkualitas rendah menimbulkan tantangan logistik, seperti pengelolaan barang rusak yang memerlukan waktu dan biaya ekstra untuk penyortiran [7]. Penjual juga menanggung risiko produk yang tidak terjual atau lebih cepat rusak saat penyimpanan, yang memperpendek masa simpan barang olahan [8]. Situasi ini berpotensi meningkatkan limbah dan kerugian finansial, karena produk menjadi tidak layak jual dan harus dibuang atau dijual dengan harga lebih rendah. Dari sisi konsumen, kualitas produk olahan yang dibuat dari kedelai



Lisensi

Lisensi Internasional Creative Commons Attribution-ShareAlike 4.0.

berkualitas rendah berdampak negatif pada pengalaman konsumen. Tempe, tahu, atau produk olahan lainnya mungkin memiliki tekstur yang tidak optimal, rasa yang kurang enak, atau masa simpan yang lebih pendek, sehingga cepat rusak [8]. Hal ini mengakibatkan ketidakpuasan konsumen dan menurunkan kepercayaan terhadap produk, yang pada akhirnya menurunkan permintaan pasar [9]. Mengingat banyaknya kerugian yang bisa terjadi, penting untuk mengembangkan sistem klasifikasi kualitas kedelai yang akurat dan otomatis, sebagai solusi signifikan untuk mengurangi kerugian di semua tingkat produksi. Sistem semacam ini memungkinkan petani untuk dengan cepat mengidentifikasi kualitas biji dan menyesuaikan proses panen serta pasca-panen guna mempertahankan kualitas optimal. Salah satu metode yang dapat diterapkan dalam pengembangan sistem klasifikasi ini adalah penggunaan teknologi pemrosesan citra digital. Dengan teknologi ini, klasifikasi kualitas kedelai dapat dilakukan secara otomatis berdasarkan karakteristik visual biji, seperti bentuk, warna, dan tekstur.

Klasifikasi citra merupakan metode penting dalam pengelompokan kualitas biji kedelai menggunakan teknik *computer vision*, yang berfokus pada pengambilan, pemrosesan, dan analisis gambar visual seperti foto atau video [10]. Tujuan dari proses klasifikasi ini adalah untuk mengidentifikasi dan mengelompokkan objek atau pola dalam gambar ke dalam kategori tertentu, seperti biji kedelai berkualitas tinggi dan berkualitas rendah [11]. Untuk mencapai akurasi yang optimal, pembelajaran mendalam (*deep learning*) diterapkan melalui Jaringan Saraf Tiruan yang memungkinkan analisis visual lebih mendalam dengan kemampuan mengenali pola kompleks yang sulit diidentifikasi secara manual [12]. Salah satu metode *deep learning* yang paling efektif digunakan dalam proses klasifikasi citra adalah *Convolutional Neural Network* (CNN), sebuah arsitektur yang dirancang untuk memproses data dua dimensi seperti gambar. CNN bekerja dengan mengekstrak fitur penting dari gambar melalui lapisan konvolusi, di mana filter atau kernel digunakan untuk mendeteksi elemen seperti tepi, tekstur, dan pola [13]. Selanjutnya, lapisan *pooling* berfungsi mengurangi dimensi peta fitur untuk mengurangi kompleksitas komputasi tanpa kehilangan informasi yang signifikan [14]. Lapisan *fully-connected* kemudian menggabungkan semua fitur yang telah dipelajari untuk membuat prediksi akhir, seperti mengklasifikasikan gambar ke dalam kategori tertentu [15][16]. Jika dataset yang digunakan terbatas, *transfer learning* dapat diterapkan. *Transfer learning* dalam CNN adalah teknik pembelajaran mesin di mana model yang sebelumnya dilatih untuk suatu tugas digunakan kembali sebagai titik awal untuk tugas baru atau tugas serupa [17]. Dengan demikian, model CNN yang telah dilatih sebelumnya pada dataset besar seperti *ImageNet* dapat dimodifikasi untuk tugas klasifikasi baru. Teknik

ini memungkinkan model memanfaatkan pengetahuan yang telah diperoleh sebelumnya untuk meningkatkan akurasi klasifikasi pada dataset yang lebih kecil dan lebih spesifik.

Dalam membangun model *deep learning*, untuk mencapai akurasi optimal tidak hanya diperlukan model yang tepat, tetapi juga dataset yang seimbang. Untuk memastikan hal ini, penyeimbangan data antar kelas sangat penting, sehingga model dapat belajar secara efektif tanpa bias. Salah satu metode yang efektif untuk mengatasi masalah ketidakseimbangan kelas dalam dataset adalah *Synthetic Minority Over-sampling Technique* (SMOTE), yang sering digunakan dalam *machine learning* untuk menangani kasus di mana satu kelas memiliki jumlah sampel yang jauh lebih sedikit dibandingkan dengan kelas lainnya. Ketidakseimbangan data ini dapat menyebabkan model lebih condong terhadap kelas mayoritas dan mengabaikan kelas minoritas. Metode SMOTE bekerja dengan mensintesis sampel baru dari kelas minoritas, sehingga meningkatkan jumlahnya dan menciptakan keseimbangan antara kelas-kelas, yang membantu model untuk belajar secara lebih adil dan efektif [18]. Penelitian ini mempresentasikan pengembangan sistem klasifikasi kualitas biji kedelai yang menggunakan SMOTE untuk menangani ketidakseimbangan data. Sistem ini mengintegrasikan kerangka kerja *deep learning*, khususnya model *transfer learning* CNN dan *MobileNetV2*, untuk meningkatkan hasil klasifikasi. CNN digunakan karena kemampuannya dalam mengekstraksi fitur visual yang kompleks dari gambar biji kedelai, sehingga model dapat membedakan kualitas biji dengan presisi tinggi. Selain itu, *MobileNetV2* dipilih karena arsitekturnya yang ringan dan efisien, memungkinkan implementasi model yang hemat sumber daya tanpa mengorbankan akurasi [19]. *Transfer learning* dari *MobileNetV2* memungkinkan model memanfaatkan fitur yang telah dipelajari dari dataset besar, lalu menyesuainya dengan tugas khusus untuk mengklasifikasikan kualitas biji kedelai. Oleh karena itu, dengan menggunakan SMOTE untuk menyeimbangkan data, diharapkan akan menghasilkan dataset yang lebih baik, membantu model memahami pola data dengan lebih efektif, dan meningkatkan kinerjanya dalam tugas klasifikasi kualitas biji kedelai.

Pada penelitian sebelumnya, telah dilakukan pembahasan mengenai proses klasifikasi kualitas biji kedelai oleh Pratap *et. all*, dalam penelitian tersebut dibangun model klasifikasi menggunakan VGG16, *AlexNet* dan CNN [20]. Hasil penelitian tersebut menunjukkan bahwa VGG16 mencapai akurasi sebesar 90%, sementara *AlexNet* memperoleh akurasi 78,49%, dan CNN mencapai akurasi 74,55%. Meskipun hasil ini sudah cukup optimal, masih terdapat ruang untuk peningkatan lebih lanjut. Oleh karena itu, dalam penelitian ini dibangun model klasifikasi kualitas biji kedelai menggunakan metode CNN dan *MobileNetV2*

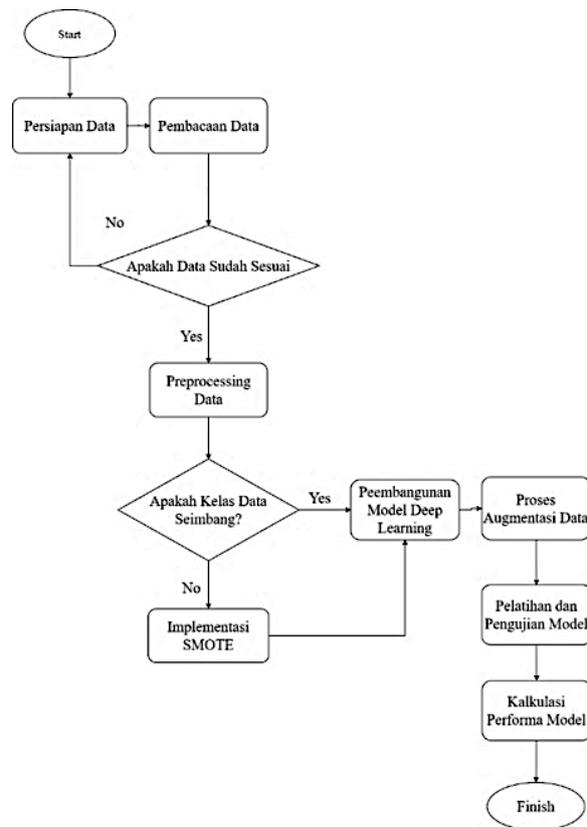
yang diperbaiki dengan penerapan SMOTE (*Synthetic Minority Over-sampling Technique*), yang di mana bertujuan untuk mempermudah dalam proses pemilahan kualitas biji kedelai. SMOTE digunakan karena distribusi data dalam dataset yang digunakan kurang seimbang, yang dapat menyebabkan model cenderung bias saat memprediksi data baru. Selain itu, penelitian ini juga memperkenalkan metode augmentasi data untuk meningkatkan generalisasi model. Augmentasi data membantu model dalam mempelajari variasi yang lebih luas dari data, seperti perbedaan sudut, pencahayaan, dan skala yang sering muncul dalam kondisi nyata. Dengan menambahkan teknik augmentasi, diharapkan model dapat lebih efektif dalam menangkap pola-pola yang ada pada citra biji kedelai, yang sebelumnya mungkin sulit dideteksi dengan model yang hanya dilatih pada data asli. Penggunaan *MobileNetV2* dalam penelitian ini memberikan keunggulan dalam hal efisiensi komputasi, karena arsitekturnya yang ringan dan cepat, namun tetap mampu mempertahankan akurasi yang tinggi. Dikombinasikan dengan SMOTE dan augmentasi data, model yang dibangun diharapkan dapat mengatasi masalah ketidakseimbangan data dan meningkatkan kemampuan model dalam menangani prediksi pada data baru dengan akurasi yang lebih optimal dibandingkan penelitian sebelumnya. Dengan pendekatan ini, diharapkan performa model dapat lebih unggul dalam mengklasifikasikan kualitas biji kedelai, baik pada data uji maupun di dunia nyata.

Berdasarkan tinjauan literatur, metode CNN terbukti unggul dalam mengekstraksi informasi dari data gambar, memungkinkan model untuk belajar secara optimal. Dalam penelitian ini, serangkaian pengujian dilakukan untuk mengevaluasi kinerja berbagai model dalam klasifikasi gambar. Hasilnya menunjukkan bahwa model CNN mencapai akurasi pengujian sebesar 91,09%, yang membuktikan kemampuannya dalam mengenali pola visual secara efektif. Namun, ketika dikombinasikan dengan teknik SMOTE untuk mengatasi ketidakseimbangan data, akurasi sedikit menurun menjadi 89,92%. Meskipun demikian, penggunaan SMOTE sangat penting untuk menangani distribusi kelas yang tidak seimbang, yang berpotensi meningkatkan generalisasi model pada kelas minoritas. Sementara itu, model *MobileNetV2*, yang merupakan arsitektur CNN yang ringan dan efisien, mencapai akurasi lebih tinggi sebesar 91,11%. Setelah penerapan SMOTE, akurasi *MobileNetV2* meningkat menjadi 92,42%, menunjukkan bahwa kombinasi SMOTE dan *transfer learning* dapat meningkatkan kinerja klasifikasi, terutama dalam konteks dataset yang tidak seimbang. Dari hasil ini dapat disimpulkan bahwa model CNN dan *MobileNetV2* yang dikembangkan dalam penelitian ini mampu mengklasifikasikan gambar dengan efektif, dan penerapan SMOTE terbukti meningkatkan akurasi model secara keseluruhan, terutama pada *MobileNetV2*. Hal ini menunjukkan

bahwa model yang dibangun bekerja dengan baik dalam tugas klasifikasi gambar dan cocok untuk digunakan dalam berbagai aplikasi yang membutuhkan presisi klasifikasi yang tinggi.

2. Metode Penelitian

Dalam penelitian ini, akan dikembangkan model *deep learning* yang dapat digunakan untuk mengklasifikasikan kualitas biji kedelai. Proses pengembangan model ini akan menggunakan bahasa pemrograman *Python*, karena *Python* menyediakan lingkungan yang sangat mendukung untuk membangun model dan implementasi sistem. *Jupyter Notebook*, sebuah platform untuk menjalankan *Python*, akan digunakan untuk tujuan ini. Alur kerja penelitian digambarkan pada gambar 1.



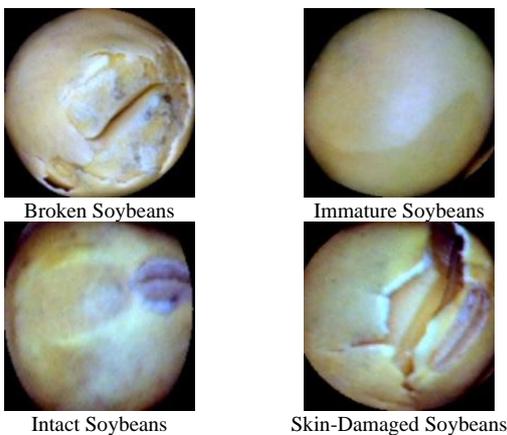
Gambar 1. Alur Penelitian

Gambar 1 menggambarkan alur kerja yang diikuti untuk membangun model klasifikasi kualitas biji kedelai menggunakan metode CNN dan *transfer learning* dengan arsitektur *MobileNetV2*. Seperti yang ditunjukkan dalam gambar 1, langkah pertama adalah mempersiapkan dataset penelitian dan membacanya. Setelah dataset dipastikan akurat, tahap praproses dimulai. Dalam penelitian ini, praproses melibatkan pengubahan ukuran dimensi piksel data. Untuk model CNN, data diubah ukurannya menjadi 150 x 150 piksel, sedangkan untuk model *MobileNetV2*, ukurannya diubah menjadi 224 x 224 piksel. Setelah praproses data selesai, langkah berikutnya adalah memeriksa

apakah distribusi data seimbang di semua kelas. Dalam penelitian ini, distribusi data antar kelas masih tidak seimbang sehingga dilakukan proses untuk menyeimbangkan distribusi data menggunakan metode SMOTE guna memastikan distribusi yang seimbang di antara kelas-kelas. Setelah pemrosesan data awal selesai, model *deep learning* dapat dibangun menggunakan CNN dan *MobileNetV2*. Setelah model siap, dilakukan augmentasi data untuk menambah variasi pada data. Terakhir, setelah semua persiapan data dan model selesai, pelatihan dan pengujian model dapat dilakukan. Setelah pengujian, kinerja model dihitung menggunakan *confusion matrix*.

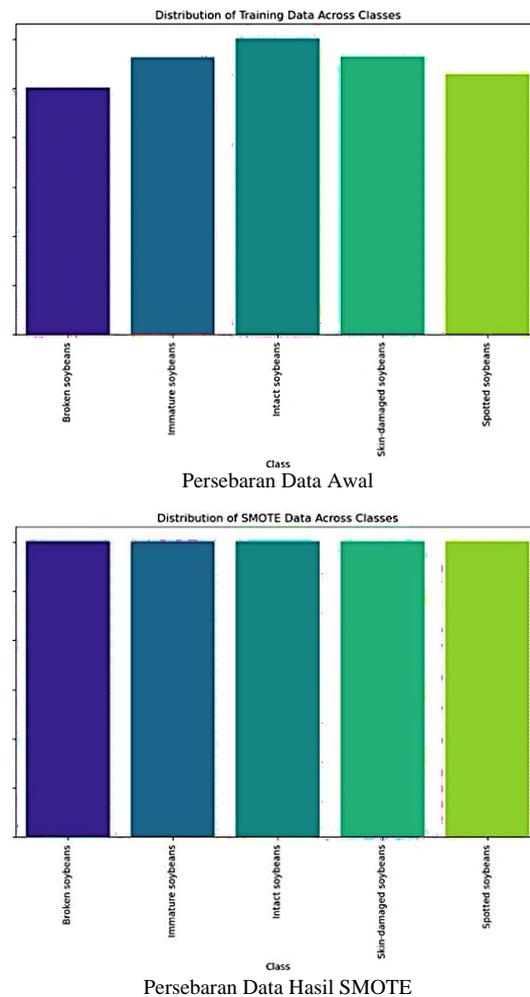
2.1. Dataset

Dalam membangun model, penting untuk memiliki dataset yang memungkinkan pelatihan model untuk mengenali pola dari data serta menguji model yang telah dikembangkan. Pada penelitian ini, dataset yang digunakan bersumber dari dataset publik dengan tautan <https://www.kaggle.com/datasets/warcoder/soyabean-seeds>. Dataset ini terdiri dari lima kelas yaitu kedelai pecah, kedelai belum matang, kedelai utuh, kedelai dengan kerusakan kulit, dan kedelai berbintik. Kedelai pecah merujuk pada biji yang retak atau rusak, sering kali akibat penanganan yang kasar, yang menunjukkan kualitas yang lebih rendah. Kedelai belum matang adalah biji yang belum mencapai kematangan penuh, sehingga tidak memenuhi standar kualitas untuk produksi yang optimal. Sebaliknya, kedelai utuh adalah biji yang masih utuh dan tidak rusak, yang mewakili kualitas tinggi, dan layak untuk dikonsumsi atau diproses. Kedelai dengan kerusakan kulit memiliki kerusakan pada bagian luar kulitnya, yang sering disebabkan oleh faktor lingkungan seperti cuaca buruk atau kesalahan selama transportasi. Terakhir, kedelai berbintik adalah biji yang memiliki bercak atau noda di permukaannya, biasanya menunjukkan adanya penyakit atau kerusakan fisik yang dapat menurunkan kualitas keseluruhan. Visualisasi data untuk setiap kelas dalam dataset ini ditampilkan pada gambar 2.



Gambar 2. Visualisasi Kelas pada Dataset

Gambar 2 menunjukkan visualisasi data dari setiap kelas dalam dataset. Pada penelitian ini, dataset yang digunakan berisi total 5.513 gambar dengan ekstensi .jpg. Namun, dataset tersebut masih menunjukkan distribusi data yang tidak merata di setiap kelas. Oleh karena itu, penelitian ini melibatkan pemrosesan dataset untuk menyeimbangkan data di setiap kelas menggunakan metode SMOTE. SMOTE bekerja dengan menghasilkan data sintesis berdasarkan data minoritas di kelas yang lebih sedikit, sehingga mencapai jumlah sampel yang seimbang di setiap kelas [21]. Visualisasi distribusi data sebelum dan sesudah proses SMOTE ditampilkan pada gambar 3.



Gambar 3. Persebaran Data Awal dan Setelah SMOTE

Gambar 3 menggambarkan distribusi data sebelum dan setelah proses SMOTE. Seperti yang ditunjukkan

dalam gambar, distribusi data awal tidak seimbang, tetapi ketika metode SMOTE diterapkan, distribusi data menjadi seimbang. Hal ini diharapkan dapat memungkinkan model untuk mempelajari pola data dengan lebih efektif dan mencapai akurasi yang optimal.

2.2. Augmentasi Data

Augmentasi data adalah teknik yang digunakan dalam pembelajaran mesin untuk memperbanyak jumlah data pelatihan dengan cara memodifikasi data yang sudah ada [22]. Tujuannya adalah untuk meningkatkan kemampuan generalisasi model dan mengurangi risiko *overfitting*, terutama ketika jumlah data yang tersedia terbatas. Untuk parameter data augmentasi yang digunakan diberikan pada tabel 1.

Tabel 1. Parameter Data Augmentasi

Parameter	CNN	MobileNetV2
<i>Rotation range</i>	30	40
<i>Width shift range</i>	0.2	0.3
<i>Height shift range</i>	0.2	0.3
<i>Shear range</i>	0.2	0.3
<i>Zoom range</i>	0.2	0.3
<i>Horizontal flip</i>	True	True
<i>Fill mode</i>	Nearest	Nearest

Tabel 1 menunjukkan parameter augmentasi data yang digunakan. Dapat dilihat pada tabel 1, model CNN dan *MobileNetV2* memiliki parameter augmentasi data yang serupa, namun terdapat beberapa perbedaan yang signifikan. Pada *rotation range*, *MobileNetV2* memiliki kisaran yang lebih besar 40° dibandingkan CNN 30°, yang menunjukkan bahwa *MobileNetV2* lebih agresif dalam menangani rotasi gambar. *Width shift range* dan *height shift range* pada *MobileNetV2* juga lebih besar (0,3) dibandingkan dengan CNN (0,2), yang berarti *MobileNetV2* menerapkan pergeseran gambar secara *horizontal* dan *vertikal* dalam skala yang lebih besar. *Shear range* dan *zoom range* pada *MobileNetV2* juga lebih luas (0,3) dibandingkan CNN (0,2), menunjukkan bahwa *MobileNetV2* memperkenalkan lebih banyak distorsi geser dan pembesaran selama augmentasi. Keduanya mengizinkan *horizontal flip* untuk memperluas keragaman sudut pandang gambar dan *fill mode* yang digunakan pada kedua model adalah *nearest*, di mana piksel kosong yang muncul akibat transformasi diisi dengan nilai piksel terdekat.

2.3. Convolutional Neural Network

CNN adalah metode jaringan saraf tiruan yang dirancang untuk memproses data yang memiliki struktur *grid*, seperti data gambar. Metode ini sangat efektif dalam tugas-tugas seperti deteksi objek, pemrosesan gambar, dan pengenalan wajah [23]. Arsitektur CNN biasanya terdiri dari tiga jenis lapisan utama, yaitu lapisan konvolusi, lapisan *pooling* dan lapisan *fully-connected*. Pada lapisan konvolusi, filter berukuran tetap, yang juga dikenal sebagai *kernel*, bergerak melintasi input. Pergerakan ini menghitung proses perkalian titik (*dot product*) antara elemen-

elemen input dan kernel, menghasilkan keluaran yang disebut sebagai *feature map* [15]. Rumus untuk lapisan konvolusi disediakan di poin 1.

$$S(i, j) = \sum_m \sum_n I(i + m, j + n) \cdot K(m, n) \quad (1)$$

Poin 1 menjelaskan perhitungan matematis untuk lapisan konvolusi. Seperti yang ditunjukkan pada poin 1, $S(i, j)$ merepresentasikan nilai pada posisi (i, j) di keluaran peta fitur (*feature map*), di mana I adalah input dan K adalah kernel. *Pooling* berfungsi untuk mengurangi dimensi dari peta fitur, biasanya menggunakan *max-pooling*, yang memilih nilai maksimum dari setiap area kecil di peta fitur [14]. Jika *max-pooling* dengan ukuran 2x2 digunakan, perhitungan untuk lapisan *pooling* disediakan di poin 2.

$$P(i, j) = \max(S(2i, 2j), S(2i + 1, 2j), S(2i, 2j + 1), S(2i + 1, 2j + 1)) \quad (2)$$

Poin 2 menjelaskan perhitungan matematis untuk lapisan *pooling*. Seperti yang ditunjukkan pada poin 2, $P(i, j)$ merepresentasikan nilai maksimum dari area 2x2 di peta fitur (S). Akhirnya, lapisan *fully-connected* menghubungkan *neuron* yang dihasilkan di lapisan input dengan *neuron* di lapisan output, mirip dengan jaringan saraf tiruan *feed forward* tradisional [16]. Rumus yang digunakan untuk lapisan *fully-connected* disediakan di poin 3.

$$O_k = f \left(\sum_i W_{ki} \cdot x_i + b_k \right) \quad (3)$$

Poin 3 menjelaskan perhitungan matematis untuk lapisan *fully-connected*. Seperti yang ditunjukkan di Poin 3, O_k merepresentasikan output dari nilai *neuron* ke- k , kemudian x_i adalah input, W_{ki} adalah bobot, b_k adalah bias, dan f merujuk pada fungsi aktivasi. Lapisan-lapisan ini bekerja bersama, memungkinkan CNN secara otomatis mengekstraksi dan mengklasifikasikan fitur kompleks dari data input. Untuk layer CNN yang digunakan pada penelitian ini diberikan pada Gambar 4.

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 158, 158, 64)	1,792
max_pooling2d (MaxPooling2D)	(None, 75, 75, 64)	0
batch_normalization (BatchNormalization)	(None, 75, 75, 64)	256
conv2d_1 (Conv2D)	(None, 75, 75, 128)	73,856
max_pooling2d_1 (MaxPooling2D)	(None, 37, 37, 128)	0
batch_normalization_1 (BatchNormalization)	(None, 37, 37, 128)	512
conv2d_2 (Conv2D)	(None, 37, 37, 256)	295,168
max_pooling2d_2 (MaxPooling2D)	(None, 18, 18, 256)	0
batch_normalization_2 (BatchNormalization)	(None, 18, 18, 256)	1,024
conv2d_3 (Conv2D)	(None, 18, 18, 512)	1,180,160
max_pooling2d_3 (MaxPooling2D)	(None, 9, 9, 512)	0
batch_normalization_3 (BatchNormalization)	(None, 9, 9, 512)	2,048
conv2d_4 (Conv2D)	(None, 9, 9, 512)	2,359,808
max_pooling2d_4 (MaxPooling2D)	(None, 4, 4, 512)	0
batch_normalization_4 (BatchNormalization)	(None, 4, 4, 512)	2,048
conv2d_5 (Conv2D)	(None, 4, 4, 1024)	4,719,616
max_pooling2d_5 (MaxPooling2D)	(None, 2, 2, 1024)	0
batch_normalization_5 (BatchNormalization)	(None, 2, 2, 1024)	4,096
flatten (Flatten)	(None, 4096)	0
dense (Dense)	(None, 2048)	8,390,656
dropout (Dropout)	(None, 2048)	0
dense_1 (Dense)	(None, 1024)	2,098,176
dropout_1 (Dropout)	(None, 1024)	0
dense_2 (Dense)	(None, 512)	524,800
dropout_2 (Dropout)	(None, 512)	0
dense_3 (Dense)	(None, 5)	2,565

Total params: 19,656,581 (74.98 MB)
 Trainable params: 19,651,589 (74.96 MB)
 Non-trainable params: 4,992 (19.50 KB)

Gambar 4. Layer CNN

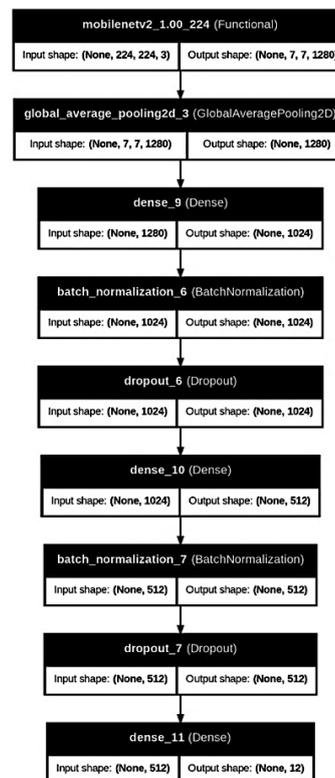
Gambar 4 menunjukkan layer CNN yang digunakan. Dapat dilihat pada gambar 4, model CNN yang digunakan memiliki arsitektur yang kompleks dengan 19.656.581 parameter, terdiri dari lapisan konvolusi bertingkat dengan jumlah filter yang meningkat dari 64 hingga 1024 untuk mengekstraksi fitur gambar secara bertahap. Setiap lapisan konvolusi diikuti oleh *MaxPooling2D* untuk mengurangi dimensi spasial, serta *Batch Normalization* untuk stabilisasi pelatihan dan mempercepat konvergensi. Setelah lapisan konvolusi, model di *flatten* untuk dihubungkan ke lapisan *fully-connected (Dense)* yang dimulai dengan 2.048 unit hingga output akhir dengan 5 unit, menunjukkan bahwa model ini digunakan untuk klasifikasi 5 kelas. Penggunaan beberapa *Dropout* bertujuan untuk mencegah *overfitting*, terutama mengingat ukuran dan kompleksitas model. Kombinasi teknik ini menunjukkan bahwa model dirancang untuk mengatasi tugas klasifikasi gambar yang membutuhkan ekstraksi fitur dalam dan generalisasi yang baik.

2.4. Transfer Learning CNN MobileNetV2

Transfer learning dalam CNN adalah teknik di mana model yang sebelumnya telah dilatih menggunakan dataset besar, seperti *ImageNet* digunakan kembali untuk menyelesaikan tugas pada dataset yang berbeda dengan jumlah data yang lebih sedikit [24]. *Transfer learning* memanfaatkan kemampuan model yang telah dilatih untuk menangkap dan menerapkan fitur-fitur yang efektif dalam mengidentifikasi pola visual umum, seperti tepi, bentuk, dan tekstur. Dalam praktiknya, lapisan awal dari model pra-latih, yang berfungsi sebagai ekstraktor fitur, biasanya dibekukan, sementara

lapisan akhir, yang lebih spesifik untuk tugas tertentu, disesuaikan atau ditambahkan lapisan baru untuk mengklasifikasikan kelas pada dataset baru. Proses ini memungkinkan model baru mencapai performa tinggi dengan lebih cepat dan efisien, karena hanya sedikit parameter yang perlu dilatih dari awal. *Transfer learning* sangat efektif ketika dataset target relatif kecil atau ketika sumber daya komputasi terbatas.

MobileNetV2 adalah arsitektur CNN yang dirancang untuk efisiensi tinggi pada perangkat dengan sumber daya terbatas, seperti ponsel atau perangkat tertanam [25]. Arsitektur ini menggunakan blok bangunan dasar yang disebut "*inverted residuals*" dengan "*linear bottlenecks*". Blok ini terdiri dari dua tahap: pada tahap pertama, dimensi input diperbesar menggunakan konvolusi 1x1 (yang meningkatkan dimensi input), diikuti dengan konvolusi *depthwise* 3x3 (yang menerapkan konvolusi secara terpisah untuk setiap saluran), dan akhirnya dimensi dikurangi lagi dengan konvolusi 1x1 linear tanpa fungsi aktivasi. Pendekatan ini memungkinkan pengurangan signifikan dalam jumlah parameter dan perhitungan yang lebih cepat dibandingkan dengan arsitektur konvensional. *MobileNetV2* juga menggunakan teknik seperti ReLU6 sebagai fungsi aktivasi dan menerapkan strides dalam konvolusi untuk mengurangi resolusi spasial sambil mempertahankan performa klasifikasi yang tinggi. Arsitektur ini terbukti sangat efektif untuk aplikasi *real-time* pada perangkat mobile dan *edge computing*, di mana efisiensi sangat penting. Model *MobileNetV2* yang digunakan digambarkan dalam gambar 5 berikut.



Gambar 5. Layer MobileNetV2

2.5. *Syntetic Minority Over-Sampling Technique*

SMOTE adalah metode yang digunakan untuk mengatasi masalah ketidakseimbangan kelas dalam data pelatihan dengan cara menghasilkan contoh sintetik untuk kelas minoritas [26]. SMOTE bekerja dengan memilih titik data dari kelas minoritas dan membuat contoh baru melalui interpolasi linier antara titik data yang ada. Secara khusus, SMOTE mengidentifikasi tetangga terdekat (k) untuk setiap titik data minoritas, lalu menghasilkan titik data sintetik dengan secara acak memilih lokasi di sepanjang segmen yang menghubungkan titik yang ada dengan tetangga terdekatnya. Perhitungan dasar untuk SMOTE dijelaskan di point 4.

$$x_{new} = x_i + \lambda \cdot (x_j - x_i) \tag{4}$$

Poin 4 menjelaskan rumus SMOTE, di mana x_i adalah titik data asli dari kelas minoritas, x_j adalah titik tetangga terdekat dari x_i , dan λ adalah nilai acak antara 0 dan 1 yang menentukan seberapa jauh titik sintesis x_{new} akan berada dari x_i menuju x_j . Dengan menambahkan jumlah contoh dari kelas minoritas, SMOTE membantu model pembelajaran mesin untuk mempelajari data dari setiap kelas dengan lebih seimbang, sehingga performa model dapat meningkat pada kelas minoritas yang sebelumnya kurang terwakili.

2.6. *Confusion Matrix*

Confusion matrix adalah alat yang sangat berharga dalam proses evaluasi kinerja model klasifikasi dengan membandingkan hasil prediksi dengan label sebenarnya [27]. Matriks ini terdiri dari empat komponen utama yaitu *True Positives* (TP), *False Positives* (FP), *True Negatives* (TN), dan *False Negatives* (FN). Dengan menggunakan komponen-komponen ini, kita dapat menghitung beberapa metrik evaluasi penting seperti presisi, *recall*, dan skor F1. Presisi secara khusus mengukur rasio prediksi positif yang benar, sehingga menjadi sangat penting dalam situasi di mana *false positive* membawa biaya yang signifikan. Perhitungan presisi dijelaskan dalam rumus yang tercantum pada poin 5.

$$\text{Presisi} = \frac{TP}{TP + FP} \tag{5}$$

Recall sering disebut sebagai sensitivitas atau tingkat positif sejati (*True Positive Rate*), mengevaluasi persentase kasus positif yang sebenarnya dapat diidentifikasi dengan akurat oleh model. Metrik ini sangat berharga dalam situasi di mana tujuannya adalah untuk memastikan bahwa semua *instance* positif terdeteksi, bahkan dengan mengorbankan peningkatan *false positive*. Perhitungan *recall* mengikuti rumus yang tercantum pada poin 6.

$$\text{Recall} = \frac{TP}{TP + FN} \tag{6}$$

F1-Score merupakan nilai harmonis dari rata-rata *precision* dan *recall*, sehingga menjadi metrik penting saat berusaha mencapai keseimbangan antara kedua ukuran ini. Nilai *F1-Score* yang tinggi menunjukkan bahwa model berfungsi dengan baik dalam hal *precision* dan *recall*. Perhitungan *F1-Score* mengikuti rumus yang tercantum pada poin 7.

$$F1 - \text{Score} = 2 \times \frac{\text{Presisi} \times \text{Recall}}{\text{Presisi} + \text{Recall}} \tag{7}$$

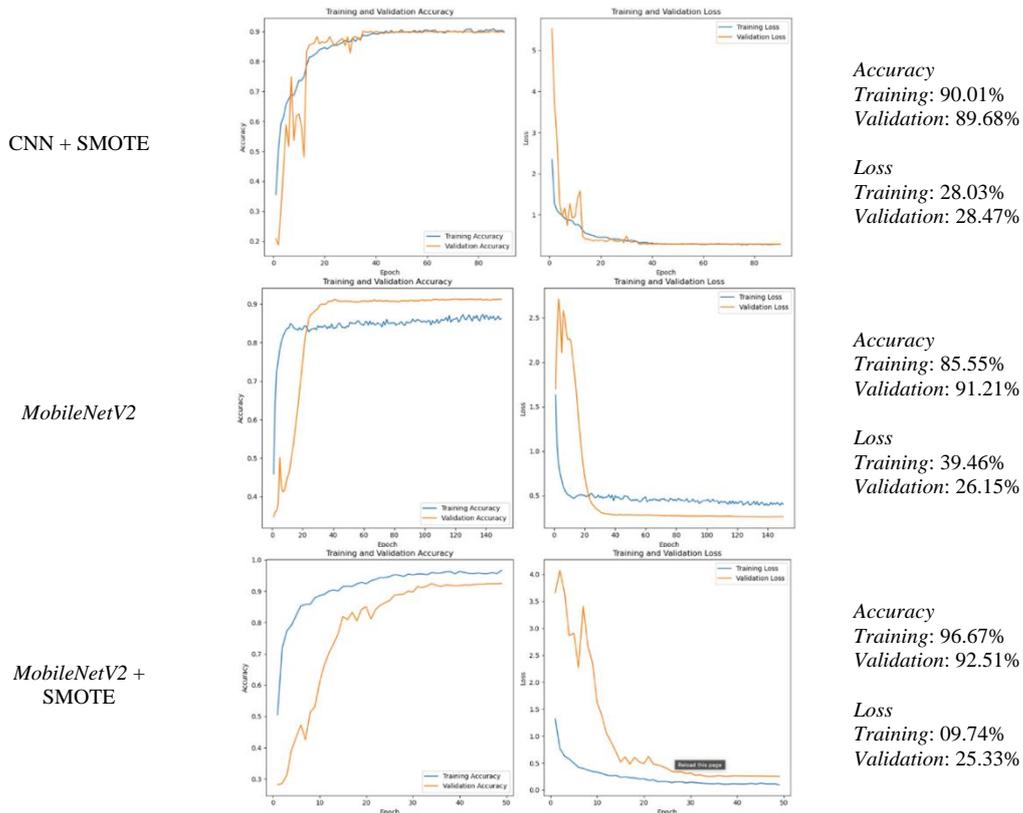
Pada dasarnya, *F1-score* mengintegrasikan kedua metrik tersebut menjadi satu nilai, memberikan penilaian yang lebih komprehensif terhadap efektivitas model. Ketiga metrik ini bekerja sama untuk memberikan wawasan berharga dalam menilai dan meningkatkan model klasifikasi.

3. Hasil dan Pembahasan

Dalam penelitian ini, *deep learning* akan dikembangkan untuk mengevaluasi kualitas biji kedelai menggunakan metodologi CNN dan *transfer learning* CNN menggunakan kerangka kerja *MobileNetV2*. Untuk meningkatkan variasi dataset, teknik SMOTE digunakan untuk praproses data bersama dengan augmentasi data untuk meningkatkan variabilitas masing-masing citra pada setiap kelas. Setelah proses augmentasi data, proses pelatihan model dilakukan. Tujuan dari pelatihan ini adalah untuk memungkinkan model mengenali pola dalam data. Setelah proses pelatihan, hasil dari pelatihan dan validasi model disajikan dalam Tabel 2.

Tabel 2. Hasil Pelatihan dan Validasi Model

Model	Grafik	Data
CNN		<p><i>Accuracy</i> Training: 91.37% Validation: 91.6%</p> <p><i>Loss</i> Training: 26.3% Validation: 26.75%</p>



Tabel 2 merangkum hasil nilai pelatihan dan nilai validasi dari berbagai metode *deep learning* yang digunakan untuk klasifikasi data. Hasilnya menunjukkan bahwa model *MobileNetV2* + SMOTE menawarkan performa tertinggi, dengan akurasi nilai pelatihan mencapai 96,67% dan akurasi nilai validasi sebesar 92,51%. Selain itu, nilai kehilangan (*loss*) pelatihan sangat rendah pada 9,74%, sedangkan nilai kehilangan validasi tercatat pada 25,33%. Ini menunjukkan bahwa model efektif dalam membuat prediksi yang akurat pada dataset pelatihan sambil mempertahankan performa validasi yang kuat, meskipun sedikit ketidaksesuaian antara nilai kehilangan pelatihan dan validasi dapat mengindikasikan kemungkinan *overfitting*. Sebaliknya, model *MobileNetV2* tanpa SMOTE memperoleh akurasi validasi yang baik sebesar 91,21%, tetapi akurasi pelatihannya lebih rendah, yaitu 85,55%. Ini menunjukkan bahwa model ini lebih baik dalam performa pada dataset validasi dibandingkan dengan dataset pelatihan, meskipun mengalami nilai kehilangan pelatihan yang lebih tinggi sebesar 39,46% dan nilai kehilangan validasi sebesar 26,15%, mencerminkan beberapa ketidakcocokan dalam prediksi yang dihasilkan.

Model CNN konvensional mencapai akurasi nilai pelatihan sebesar 91,37% dan akurasi nilai validasi sebesar 91,6%, dengan nilai kehilangan pelatihan sebesar 26,3% dan nilai kehilangan validasi sebesar 26,75%. Meskipun akurasi terlihat memuaskan, nilai kehilangan yang relatif tinggi menunjukkan bahwa

model masih mengalami kesalahan prediksi yang signifikan pada dataset pelatihan dan validasi. Setelah menerapkan SMOTE pada model CNN, terjadi sedikit penurunan akurasi, dengan akurasi pelatihan turun menjadi 90,01% dan akurasi validasi menjadi 89,68%. Selain itu, nilai kehilangan pelatihan meningkat menjadi 28,03% dan nilai kehilangan validasi menjadi 28,47%, keduanya sedikit lebih tinggi dibandingkan model CNN standar. Ini menunjukkan bahwa penggabungan SMOTE dengan CNN tidak secara signifikan meningkatkan akurasi atau mengurangi nilai kehilangan, dan bahkan dapat sedikit mengurangi performa dibandingkan dengan CNN tanpa SMOTE. Akhirnya, temuan menunjukkan bahwa kombinasi *MobileNetV2* dengan SMOTE mencapai performa keseluruhan terbaik, terutama dalam hal akurasi pelatihan dan validasi, sambil mempertahankan nilai kehilangan yang lebih rendah selama pelatihan. Ini menunjukkan bahwa kombinasi *MobileNetV2* dan SMOTE secara efektif meningkatkan performa model pada dataset yang tidak seimbang, meskipun perlu diwaspadai untuk mencegah *overfitting*, seperti yang terlihat dari ketidaksesuaian antara nilai kehilangan pelatihan dan nilai kehilangan validasi. Setelah fase pelatihan dan validasi, pengujian model dapat dilanjutkan, dengan hasil yang dirinci dalam Tabel 3.

Tabel 3. Hasil Pengujian Model

Model	Akurasi	Presisi	Recall	F1-Score
CNN	91.09%	91%	91%	91%
CNN + SMOTE	89.92%	90%	90%	90%
<i>MobileNetV2</i>	91.11%	91%	91%	91%

<i>MobileNetV2</i> + SMOTE	92.42%	93%	92%	92%
-------------------------------	--------	-----	-----	-----

Tabel 3 yang di mana merupakan hasil pengujian model *deep learning* pada klasifikasi menggunakan beberapa arsitektur dan teknik *balancing* data menunjukkan variasi performa yang menarik. Model *Convolutional Neural Network* (CNN) dasar memberikan akurasi sebesar 91.09% dengan presisi, *recall*, dan *F1-Score* masing-masing sebesar 91%. Ini menandakan bahwa CNN cukup efektif dalam menangani klasifikasi awal, dengan keseimbangan antara kemampuan mengidentifikasi kelas yang benar dan menghindari kesalahan dalam prediksi. Kombinasi nilai presisi dan *recall* yang seimbang pada angka 91% menunjukkan bahwa CNN mampu secara konsisten memberikan performa yang baik dalam mengklasifikasikan data citra yang diberikan. Namun, ketika teknik *Synthetic Minority Over-sampling Technique* (SMOTE) diterapkan pada CNN, terdapat sedikit penurunan akurasi menjadi 89.92%. Meskipun demikian, presisi, *recall*, dan *F1-Score* tetap berada pada angka 90%. Hal ini mengindikasikan bahwa meskipun SMOTE biasanya diharapkan dapat meningkatkan performa dengan mengatasi ketidakseimbangan data, dalam kasus ini, akurasi model sedikit menurun. Namun, keseragaman antara presisi, *recall*, dan *F1-Score* yang tetap pada 90% menunjukkan bahwa model tetap mampu menjaga konsistensi performanya, meskipun akurasinya sedikit lebih rendah dibandingkan CNN dasar.

Model *MobileNetV2*, yang dioptimalkan untuk efisiensi komputasi, menunjukkan akurasi yang lebih tinggi dibandingkan dengan CNN tradisional, mencapai 91,11%. Model ini memiliki metrik presisi, *recall*, dan *F1-Score* masing-masing sebesar 91%, yang menunjukkan bahwa *MobileNetV2* sangat efektif dalam klasifikasi citra, hampir setara dengan CNN. Menariknya, meskipun tanpa penerapan SMOTE, *MobileNetV2* menunjukkan kinerja yang kuat, menyoroti keunggulan arsitekturnya dalam mengelola tugas klasifikasi tanpa bergantung pada metode *oversampling*. Ketika SMOTE digunakan bersamaan dengan *MobileNetV2*, terjadi peningkatan kinerja yang signifikan. Akurasinya naik menjadi 92,42%, dengan presisi mencapai 93%, *recall* 92%, dan *F1-Score* 92%. Peningkatan ini menunjukkan efektivitas SMOTE dalam meningkatkan kapasitas *MobileNetV2* untuk mengatasi dataset yang tidak seimbang. Kenaikan dalam presisi dan *recall* menunjukkan bahwa model ini tidak hanya meningkatkan akurasi tetapi juga menjadi lebih dapat diandalkan dalam mengidentifikasi kelas dengan benar dan mengurangi kesalahan prediksi. Secara keseluruhan, kombinasi *MobileNetV2* dengan SMOTE menghasilkan kinerja tertinggi, melampaui model lain dalam berbagai metrik. Oleh karena itu, *MobileNetV2* + SMOTE muncul sebagai pilihan terbaik untuk klasifikasi citra, terutama dalam konteks di mana

ketidakseimbangan data menjadi perhatian, sehingga menghasilkan hasil yang lebih akurat dan konsisten. Temuan ini semakin menekankan bahwa mengintegrasikan teknik penyeimbangan data seperti SMOTE dapat secara signifikan meningkatkan kinerja arsitektur canggih seperti *MobileNetV2*. Dengan akurasi yang luar biasa, serta presisi, *recall*, dan *F1-Score* yang tinggi, model *MobileNetV2* + SMOTE menonjol sebagai pilihan yang sangat baik untuk mengklasifikasikan kualitas biji kedelai dan sangat direkomendasikan untuk tugas klasifikasi yang serupa.

Telah dilakukan penelitian mengenai proses klasifikasi kualitas biji kedelai menggunakan berbagai metode. Misalnya, sebuah studi yang dirujuk dalam [28] menggunakan teknik CNN dan *transfer learning* CNN, khususnya VGG16 dan AlexNet, untuk mengklasifikasikan kualitas biji kedelai. Penelitian ini mencapai akurasi sebesar 90% dengan model VGG16, 78,49% dengan model AlexNet, dan 74,55% dengan model CNN konvensional. Studi tersebut membandingkan beberapa metode, termasuk CNN, CNN + SMOTE, *MobileNetV2*, dan *MobileNetV2* + SMOTE, dengan akurasi masing-masing sebesar 95,28%, 96,43%, 95,78%, dan 97,07%. Hasil ini menunjukkan bahwa *MobileNetV2* yang dipadukan dengan SMOTE mencapai akurasi tertinggi, menyoroti efektivitas teknik penyeimbangan data dalam meningkatkan kinerja klasifikasi citra, bahkan melampaui model yang dikembangkan dalam penelitian sebelumnya dalam konteks ini. Mengingat terbatasnya studi yang membahas klasifikasi kualitas biji kedelai, penelitian ini diharapkan menjadi referensi untuk mengembangkan model klasifikasi yang akurat dan lebih optimal.

4. Kesimpulan

Penelitian ini berfokus pada klasifikasi kualitas biji kedelai melalui penggunaan CNN dan *transfer learning*, khususnya *MobileNetV2*. Hasil pengujian menunjukkan bahwa model CNN mencapai nilai akurasi sebesar 91,09%. Ketika CNN dipadukan dengan SMOTE, akurasinya sedikit lebih rendah, yaitu 89,92%. Di sisi lain, *MobileNetV2* mencapai akurasi sebesar 91,11%, sementara kombinasi *MobileNetV2* dan SMOTE menghasilkan akurasi tertinggi sebesar 92,42%. Temuan ini menunjukkan bahwa model *MobileNetV2*, terutama ketika menggunakan teknik *preprocessing* SMOTE, memberikan akurasi terbaik dibandingkan metode lainnya dan penelitian sebelumnya. Informasi ini dapat menjadi dasar untuk mengembangkan model yang lebih tepat dalam mengklasifikasikan kualitas biji kedelai, memastikan bahwa biji yang diproses memiliki kualitas yang lebih baik dan memaksimalkan nilai gizi. Namun, masih ada potensi untuk meningkatkan akurasi yang diperoleh, dan optimasi lebih lanjut dapat dilakukan. Penelitian mendatang dapat memanfaatkan eksplorasi model yang

lebih canggih, termasuk kombinasi berbagai arsitektur, untuk meningkatkan variabilitas dalam kinerja klasifikasi kualitas biji kedelai dan mencapai akurasi yang optimal.

Daftar Rujukan

- [1] P. Qin, T. Wang, and Y. Luo, "A review on plant-based proteins from soybean: Health benefits and soy product development," *J Agric Food Res*, vol. 7, Mar. 2022, doi: 10.1016/j.jafr.2021.100265.
- [2] I. S. Kim, C. H. Kim, and W. S. Yang, "Physiologically active molecules and functional properties of soybeans in human health—a current perspective," Apr. 02, 2021, *MDPI*. doi: 10.3390/ijms22084054.
- [3] J. O. Karlsson, A. Parodi, H. H. E. van Zanten, P. A. Hansson, and E. Röös, "Halting European Union soybean feed imports favours ruminants over pigs and poultry," *Nat Food*, vol. 2, no. 1, pp. 38–46, Jan. 2021, doi: 10.1038/s43016-020-00203-7.
- [4] W. Kudelka, M. Kowalska, and M. Popis, "Quality of soybean products in terms of essential amino acids composition," *Molecules*, vol. 26, no. 16, Aug. 2021, doi: 10.3390/molecules26165071.
- [5] W. William, B. Dahl, and D. Hertsgaard, "Soybean quality differentials, blending, testing and spatial arbitrage," *Journal of Commodity Markets*, vol. 18, Jun. 2020, doi: 10.1016/j.jcomm.2019.100095.
- [6] K. Karges, S. D. Bellingrath-Kimura, C. A. Watson, F. L. Stoddard, M. Halwani, and M. Reckling, "Agro-economic prospects for expanding soybean production beyond its current northerly limit in Europe," *European Journal of Agronomy*, vol. 133, Feb. 2022, doi: 10.1016/j.eja.2021.126415.
- [7] F. A. Asodina, F. Adams, F. Nimoh, E. A. Weyori, C. A. Wongnaa, and J. E. A. Bakang, "Are non-market benefits of soybean production significant? An extended economic analysis of smallholder soybean farming in Upper West region of northern Ghana," *Agric Food Secur*, vol. 9, no. 1, Dec. 2020, doi: 10.1186/s40066-020-00265-7.
- [8] Z. Huang *et al.*, "Evaluating the effect of different processing methods on fermented soybean whey-based tofu quality, nutrition, and flavour," *LWT*, vol. 158, Mar. 2022, doi: 10.1016/j.lwt.2022.113139.
- [9] M. Lihoussou and S. Limbourg, "Towards a sustainable production of maize and soybean in the department of Borgou," *Cleaner Logistics and Supply Chain*, vol. 4, Jul. 2022, doi: 10.1016/j.clscn.2022.100039.
- [10] L. Chen, S. Li, Q. Bai, J. Yang, S. Jiang, and Y. Miao, "Review of image classification algorithms based on convolutional neural networks," Nov. 01, 2021, *MDPI*. doi: 10.3390/rs13224712.
- [11] D. Bhatt *et al.*, "Cnn variants for computer vision: History, architecture, application, challenges and future scope," Oct. 01, 2021, *MDPI*. doi: 10.3390/electronics10202470.
- [12] N. Sharma, R. Sharma, and N. Jindal, "Machine Learning and Deep Learning Applications-A Vision," *Global Transitions Proceedings*, vol. 2, no. 1, pp. 24–28, Jun. 2021, doi: 10.1016/j.gltp.2021.01.004.
- [13] M. M. Taye, "Theoretical Understanding of Convolutional Neural Network: Concepts, Architectures, Applications, Future Directions," Mar. 01, 2023, *MDPI*. doi: 10.3390/computation11030052.
- [14] A. Zafar *et al.*, "A Comparison of Pooling Methods for Convolutional Neural Networks," Sep. 01, 2022, *MDPI*. doi: 10.3390/app12178643.
- [15] X. Zhang *et al.*, "Understanding the learning mechanism of convolutional neural networks in spectral analysis," *Anal Chim Acta*, vol. 1119, pp. 41–51, Jul. 2020, doi: 10.1016/j.aca.2020.03.055.
- [16] S. H. S. Basha, S. R. Dubey, V. Pulabaigari, and S. Mukherjee, "Impact of fully connected layers on performance of convolutional neural networks for image classification," *Neurocomputing*, vol. 378, pp. 112–119, Feb. 2020, doi: 10.1016/j.neucom.2019.10.008.
- [17] M. Bansal, M. Kumar, M. Sachdeva, and A. Mittal, "Transfer learning for image classification using VGG19: Caltech-101 image data set," *J Ambient Intell Humaniz Comput*, vol. 14, no. 4, pp. 3609–3620, Apr. 2023, doi: 10.1007/s12652-021-03488-z.
- [18] R. A. Prasojo *et al.*, "Precise transformer fault diagnosis via random forest model enhanced by synthetic minority over-sampling technique," *Electric Power Systems Research*, vol. 220, Jul. 2023, doi: 10.1016/j.epr.2023.109361.
- [19] R. Indraswari, R. Rokhana, and W. Herulambang, "Melanoma image classification based on MobileNetV2 network," in *Procedia Computer Science*, Elsevier B.V., 2021, pp. 198–207. doi: 10.1016/j.procs.2021.12.132.
- [20] H. Pratap, M. S. G. Prasad, P. Agarwal, A. Bhardwaj, A. Mehra, and S. Mathpati, "SoyNet: Deep Learning Approaches for Automated Soybean Seed Quality Assessment," in *2024 2nd International Conference on Advancement in Computation & Computer Technologies (InCACCT)*, Gharuan: IEEE, Jun. 2024, pp. 613–617. doi: 10.1109/InCACCT61598.2024.10551180.
- [21] E. Erlin, Y. Desnelita, N. Nasution, L. Suryati, and F. Zoromi, "Dampak SMOTE terhadap Kinerja Random Forest Classifier berdasarkan Data Tidak seimbang," *MATRIK: Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 21, no. 3, pp. 677–690, Jul. 2022, doi: 10.30812/matrik.v21i3.1726.
- [22] M. Toyib, T. Decky, and K. Pratama, "Ilmu pengetahuan Alam," *Kebumihan dan Angkasa*, vol. 2, no. 3, pp. 108–120, 2024, doi: 10.62383/algorithm.v2i3.69.
- [23] Y. Tian, "Artificial Intelligence Image Recognition Method Based on Convolutional Neural Network Algorithm," *IEEE Access*, vol. 8, pp. 125731–125744, 2020, doi: 10.1109/ACCESS.2020.3006097.
- [24] L. Alzubaidi *et al.*, "Towards a better understanding of transfer learning for medical imaging: A case study," *Applied Sciences (Switzerland)*, vol. 10, no. 13, Jul. 2020, doi: 10.3390/app10134523.
- [25] Y. Gulzar, "Fruit Image Classification Model Based on MobileNetV2 with Deep Transfer Learning Technique," *Sustainability (Switzerland)*, vol. 15, no. 3, Feb. 2023, doi: 10.3390/su15031906.
- [26] D. Elreedy, A. F. Atiya, and F. Kamalov, "A theoretical distribution analysis of synthetic minority oversampling technique (SMOTE) for imbalanced learning," *Mach Learn*, vol. 113, no. 7, pp. 4903–4923, Jul. 2024, doi: 10.1007/s10994-022-06296-4.
- [27] A. Syahril *et al.*, "Perbandingan Metode Decision Tree Dan K-Nearest Neighbor Terhadap Ulasan Pengguna Aplikasi Mypertamina Menggunakan Confusion Matrix," *Journal of Information System Research (JOSH)*, vol. 5, no. 4, pp. 1085–1094, 2024, doi: 10.47065/josh.v5i4.5639.