

HYBRID MOBILENETV2 DAN EXTREME GRADIENT BOOSTING UNTUK KLASIFIKASI KERUSAKAN BANGUNAN

Muammar Najmi Suryanooradja^{1*}, Ani Dijah Rahajoe², Achmad Junaidi³

^{1,2,3}Fakultas Ilmu Komputer / Universitas Pembangunan Nasional Veteran Jawa Timur; Jl. Rungkut Madya No.1, Gn. Anyar. Kec. Gn, Anyar, Kota Surabaya, Jawa Timur 60294; telp +62 (031) 870 6369

Keywords:

MobileNetV2, XGBoost, feature extraction, image classification, building damage, hybrid model.

Correspondent Email:

21081010208@student.upnja tim.ac.id

Abstrak. Penelitian ini menggunakan pendekatan hybrid dengan menggabungkan MobileNetV2 sebagai metode ekstraksi fitur dan XGBoost sebagai algoritma klasifikasi. MobileNetV2 digunakan untuk mengekstraksi representasi fitur yang ringan namun kaya informasi dari citra, sementara XGBoost memanfaatkan fitur tersebut untuk membangun model klasifikasi yang akurat dan efisien. Dataset dibagi menggunakan beberapa skema pembagian data, yaitu Kaggle 80:10:10, Kaggle 70:20:10, Kaggle 60:20:20, serta dataset Mandiri dengan skema yang serupa. Model dikembangkan untuk mengklasifikasikan citra ke dalam tiga kategori kelas, yaitu *Ringan*, *Sedang*, dan *Berat*. Proses pelatihan model XGBoost membutuhkan waktu sekitar 60 detik, sedangkan proses evaluasi hanya memerlukan waktu sekitar 0,04 detik, menunjukkan efisiensi komputasi yang sangat baik. Dari seluruh skema yang diujikan, hasil terbaik diperoleh pada skema Kaggle 70:20:10, dengan akurasi mencapai 94,72%, precision 94,80%, recall 94,72%, f1-score 94,71%, dan waktu komputasi sebesar 54,91 detik. Secara umum, model menunjukkan performa yang konsisten baik pada data validasi maupun data uji, dengan nilai precision, recall, dan F1-score yang tinggi di setiap kelas. Penelitian ini menunjukkan bahwa kombinasi MobileNetV2 dan XGBoost efektif dalam menghasilkan model klasifikasi citra dengan performa akurasi tinggi dan kecepatan inferensi yang sangat cepat. Pendekatan ini dapat menjadi solusi yang menjanjikan untuk berbagai aplikasi berbasis klasifikasi citra, khususnya pada skenario yang membutuhkan respons cepat dan sumber daya komputasi yang terbatas.

Abstract. This research proposes a hybrid approach by combining MobileNetV2 as a feature extraction method and XGBoost as the classification algorithm. MobileNetV2 is utilized to extract lightweight yet information-rich feature representations from images, while XGBoost leverages these features to build an accurate and efficient classification model. The dataset was split using several data splitting schemes, namely Kaggle 80:10:10, Kaggle 70:20:10, Kaggle 60:20:20, and a custom dataset with a similar scheme. The model was developed to classify images into three class categories Minor, Moderate, and Severe damage. The training process of the XGBoost model required approximately 60 seconds, while the evaluation process only took about 0.04 seconds, demonstrating excellent computational efficiency. Among all the tested schemes, the best results were achieved with the Kaggle 70:20:10 scheme, reaching an accuracy of 94.72%, precision of 94.80%, recall of 94.72%, F1-score of 94.71%, and a computation time of 54.91 seconds. Overall, the model exhibited consistent performance on both validation and test datasets, achieving high precision, recall, and F1-score across all classes. This study demonstrates that the combination of

MobileNetV2 and XGBoost is effective in producing an image classification model with high accuracy and very fast inference speed. This approach could offer a promising solution for various image classification applications, especially in scenarios that require fast response times and limited computational resources.

1. PENDAHULUAN

Kerusakan pada bangunan merupakan persoalan serius yang berdampak langsung pada keselamatan dan kenyamanan masyarakat. Faktor-faktor seperti penyusutan, kelelahan material, dan perbedaan suhu menyebabkan penurunan kualitas bangunan seiring waktu [1]. Selama ini, proses deteksi kerusakan dilakukan secara manual oleh tenaga ahli, yang memerlukan biaya besar dan waktu yang lama [2]. Untuk menjawab tantangan tersebut, penelitian ini mengusulkan metode berbasis *deep learning* dan *machine learning* dengan menggabungkan MobileNetV2 sebagai ekstraktor fitur dan XGBoost sebagai pengklasifikasi.

MobileNetV2 merupakan arsitektur CNN efisien yang dirancang untuk perangkat dengan keterbatasan komputasi, dan terbukti efektif dalam mengenali pola visual kerusakan seperti retakan dan keruntuhan struktural [3]. Namun, untuk menentukan tingkat keparahan kerusakan, dibutuhkan klasifikasi lanjutan. Di sinilah XGBoost berperan sebagai algoritma *gradient boosting* berbasis pohon keputusan yang mampu meningkatkan akurasi klasifikasi dan mengatasi *overfitting* [4].

Kombinasi keduanya diharapkan dapat menyederhanakan proses identifikasi kerusakan bangunan menjadi tiga tingkat, yaitu ringan, sedang, dan berat, dengan akurasi yang tinggi serta efisiensi komputasi. Tantangan terbesar dalam pengembangan sistem ini adalah keterbatasan dataset kerusakan bangunan yang bervariasi dan representatif. MobileNetV2 dan XGBoost menawarkan keunggulan dalam menghadapi masalah tersebut, dengan kemampuan ekstraksi fitur dari data terbatas dan mekanisme pembelajaran berkelanjutan dari kesalahan klasifikasi [3][4].

Beberapa penelitian sebelumnya mendukung efektivitas kedua metode tersebut. Faisal dkk. (2021) menggunakan MobileNetV2 untuk klasifikasi wajah bermasker dengan akurasi 95% [5], sementara Nasha dan Ihsan (2021) mengaplikasikan model yang sama

dalam penerjemahan bahasa isyarat [6]. Di sisi lain, XGBoost telah digunakan secara efektif dalam klasifikasi kebakaran hutan oleh Karo (2020), dengan akurasi mencapai 89,52% serta nilai sensitivitas dan spesifisitas yang tinggi [7].

Melalui penelitian ini, diharapkan sistem berbasis MobileNetV2 dan XGBoost dapat menjadi alternatif modern dalam deteksi dan klasifikasi kerusakan bangunan, mengurangi ketergantungan pada inspeksi manual, serta memberikan kontribusi nyata dalam pemeliharaan infrastruktur bangunan di Indonesia.

2. TINJAUAN PUSTAKA

2.1 Kerusakan Bangunan

Dalam Peraturan Menteri PUPR Nomor 24 Tahun 2008, kerusakan bangunan dijelaskan sebagai kondisi di mana suatu bangunan atau bagian-bagiannya tidak lagi berfungsi sebagaimana mestinya akibat berbagai faktor, seperti beban berlebih, gempa bumi, kebakaran, guncangan kendaraan besar, maupun sebab-sebab lainnya. Berdasarkan tingkat keparahannya, kerusakan bangunan dibagi menjadi tiga kategori utama.

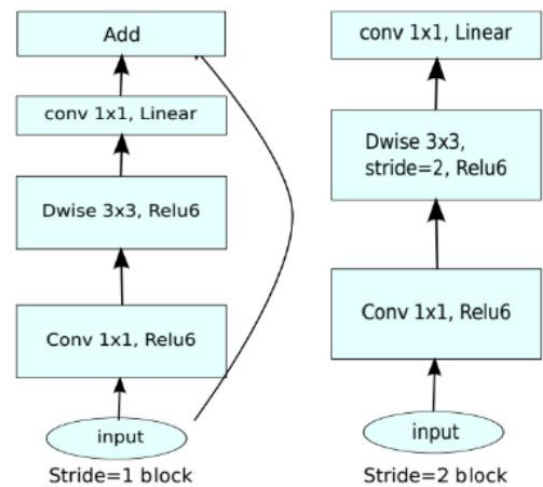
Kerusakan ringan umumnya terjadi pada elemen non-struktural seperti atap, plafon, lantai, atau dinding pengisi. Kerusakan sedang melibatkan sebagian elemen struktural maupun non-struktural. Misalnya, dinding partisi mengalami retak tembus, atau bagian struktural seperti kolom dan kuda-kuda mulai menunjukkan kerusakan, meskipun tidak sampai menyebabkan keruntuhan bangunan. Sementara itu, kerusakan berat mengindikasikan bahwa sebagian besar komponen bangunan telah rusak parah, baik pada struktur utama maupun bagian pendukungnya. Tanda-tandanya bisa berupa dinding yang miring atau retak besar, serta kerusakan serius pada kolom atau balok yang menyebabkan bangunan berisiko roboh.

2.2 Pengolahan Citra

Pengolahan citra merupakan salah satu teknik yang sering dimanfaatkan dalam mendeteksi kerusakan pada bangunan. Dengan pendekatan ini, analisis terhadap kondisi bangunan dapat dilakukan melalui gambar visual, baik itu pada bangunan tempat tinggal, tempat ibadah, bangunan usaha, fasilitas sosial budaya, maupun fungsi lainnya. Menurut RD. Kusumanto dan Alan Novi Tompunu [10], citra secara matematis didefinisikan sebagai fungsi kontinu dari intensitas cahaya pada bidang dua dimensi. Namun, agar dapat diproses oleh komputer, citra tersebut harus diubah ke dalam bentuk digital, yaitu dengan mempresentasikannya sebagai nilai-nilai diskrit. Proses ini disebut digitalisasi citra. Hasil dari digitalisasi adalah citra digital yang dapat direpresentasikan dalam bentuk matriks dua dimensi $f(x,y)$, dengan jumlah M kolom dan N baris. Setiap titik temu antara baris dan kolom membentuk elemen terkecil dari citra, yang disebut piksel (pixel atau *picture element*) [13].

2.3 MobileNetV2

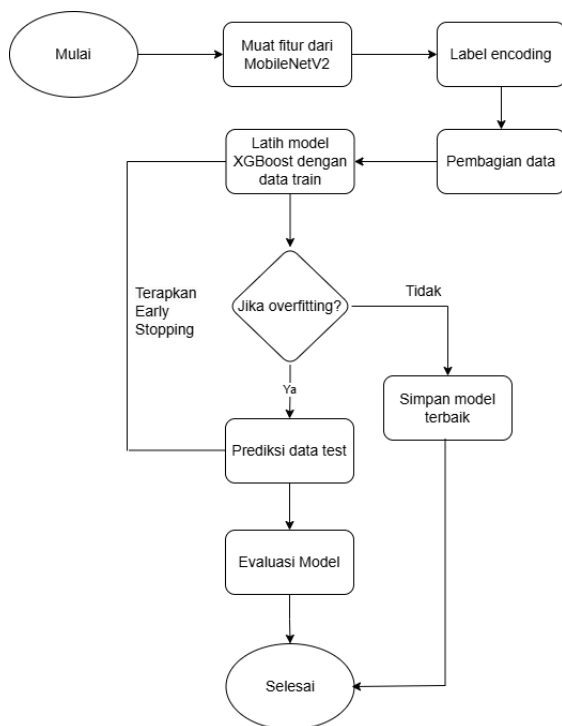
MobileNetV2 dirancang untuk mengenali fitur penting dari gambar secara efisien, bahkan ketika jumlah data yang tersedia terbatas. Model ini menggunakan *Depthwise Separable Convolution*, sebuah teknik yang mampu mengurangi beban komputasi tanpa menurunkan akurasi secara signifikan. Selain itu, arsitekturnya mengandalkan *Inverted Residual* dan *Linear Bottleneck*, yang memungkinkan ekstraksi fitur dengan parameter yang lebih sedikit namun tetap efektif [3]. Dalam strukturnya, terdapat dua jenis blok utama. Blok dengan *stride* 1 memungkinkan adanya *shortcut connection* dan terdiri dari tiga tahap: *1x1 convolution* untuk memperbesar dimensi fitur, *3x3 depthwise convolution* untuk ekstraksi fitur per channel, dan *1x1 convolution* dengan aktivasi linear untuk mengembalikan ukuran channel. Sementara itu, blok dengan *stride* 2 digunakan untuk *downsampling* atau pengurangan resolusi [11]. Blok ini memiliki struktur serupa, namun tanpa shortcut karena perbedaan ukuran input dan output. Meskipun lebih ringan, arsitektur ini tetap mampu menangkap informasi penting dari gambar secara luas dan mendalam.



Gambar 1 Arsitektur MobileNetV2

2.4 Extreme Gradient Boosting

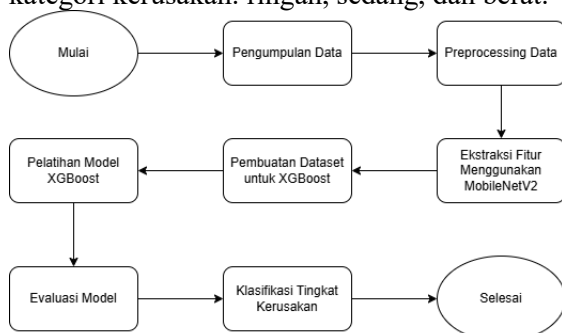
Extreme Gradient Boosting, atau XGBoost, adalah algoritma pembelajaran mesin yang dikenal karena kekuatannya dalam menangani sistem kompleks secara cepat dan akurat. Menurut Zeravan et al. (2023) [14], XGBoost dirancang agar efisien, fleksibel, dan portabel, serta sangat unggul dalam hal akurasi prediksi, interpretabilitas, dan kemampuan klasifikasinya. Algoritma ini bekerja berdasarkan pendekatan *gradient boosting* dan berfokus pada *decision tree* yang dibangun secara paralel. Dalam bidang pengolahan citra, XGBoost sering dimanfaatkan sebagai metode klasifikasi untuk menentukan tingkat kerusakan bangunan berdasarkan fitur visual yang sebelumnya telah diekstraksi dari gambar.



Gambar 2 Alur XGBoost

2.5 Hybrid Model

Penelitian ini menggabungkan MobileNetV2 dan XGBoost untuk mendeteksi serta mengklasifikasikan kerusakan bangunan berdasarkan citra. Tahapan dimulai dari pengumpulan gambar kerusakan, dilanjutkan dengan preprocessing seperti resize, normalisasi, dan augmentasi untuk memperkaya dataset. Gambar yang telah diproses kemudian dimasukkan ke dalam MobileNetV2 guna mengekstraksi fitur penting. Hasil ekstraksi ini selanjutnya menjadi input bagi model XGBoost yang bertugas mengklasifikasikan gambar ke dalam tiga kategori kerusakan: ringan, sedang, dan berat.



Gambar 3 Hybrid Model

2.6 Graphical User Interface

Graphical User Interface (GUI) dalam penelitian ini dirancang untuk mempermudah pengguna dalam mengakses sistem klasifikasi tanpa perlu menjalankan perintah teks. Antarmuka dibangun menggunakan Tkinter dan menampilkan elemen-elemen sederhana seperti tombol pemilihan gambar, area tampilan gambar, serta label hasil prediksi. Saat pengguna memilih gambar, sistem akan otomatis melakukan resize dan menyesuaikan format sesuai kebutuhan MobileNetV2. Fitur dari gambar kemudian diekstraksi dan dikirim ke model XGBoost untuk menentukan kelas kerusakan: ringan, sedang, atau berat. Hasil prediksi ditampilkan langsung di GUI, yang juga dilengkapi dengan label encoder untuk mengubah output numerik menjadi label kelas. Dengan otomatisasi penuh di balik tampilannya, GUI ini tidak hanya mempermudah penggunaan oleh pengguna non-teknis, tetapi juga memperkuat penerapan nyata dari sistem klasifikasi yang dikembangkan.

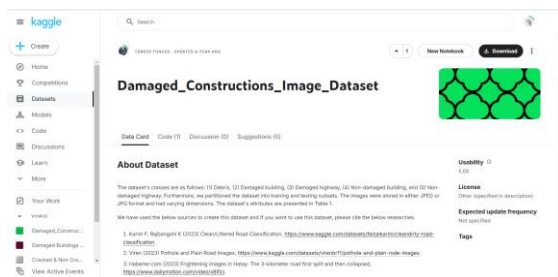
3. METODE PENELITIAN

3.1 Perancangan Sistem

Penelitian ini dirancang untuk mengklasifikasikan tingkat kerusakan bangunan berdasarkan citra visual dengan pendekatan kombinasi metode ekstraksi fitur dan klasifikasi. Data diperoleh dari dua sumber, yaitu dataset publik Kaggle dan hasil dokumentasi mandiri. Setelah melalui tahap preprocessing, citra diproses menggunakan arsitektur MobileNetV2 untuk mengekstraksi fitur penting, yang selanjutnya diklasifikasikan menggunakan algoritma XGBoost ke dalam tiga kategori kerusakan. Hasil klasifikasi kemudian dievaluasi menggunakan sejumlah metrik performa untuk menilai efektivitas model, sebelum ditarik kesimpulan dan saran lanjutan.

3.2 Sumber Dataset

Penelitian ini menggunakan dataset gabungan dari dua sumber, yaitu dokumentasi pribadi dan platform Kaggle. Sebanyak 466 gambar diperoleh secara mandiri dengan memotret bangunan rusak menggunakan handphone, sementara 540 gambar lainnya diambil dari Kaggle. Seluruh gambar mewakili tiga kategori kerusakan: ringan, sedang, dan berat. Secara total, terkumpul 1.006 gambar sebagai dataset untuk proses klasifikasi.



Gambar 4 Dataset Kaggle



Gambar 5 Dataset Mandiri

3.3 Preprocessing Data

Preprocessing dalam penelitian ini bertujuan untuk meningkatkan kualitas citra sebelum dilakukan ekstraksi fitur dan klasifikasi. Tahapan ini meliputi *resize* dan normalisasi gambar agar sesuai dengan input MobileNetV2, serta augmentasi data untuk memperluas variasi dan jumlah dataset. Enam teknik augmentasi yang digunakan adalah *flipping* vertikal dan horizontal, rotasi 45°, *scaling up*, penyesuaian *brightness & contrast*, serta *shearing*. Seluruh proses ini dilakukan untuk meningkatkan kinerja model dan menghindari overfitting.



Gambar 6 Resize

```
Dimensi gambar asli: (4080, 3060, 3)
Dimensi gambar hasil resize: (224, 224, 3)
contoh nilai piksel setelah normalisasi (5x5 pertama):
[[0.1372549 0.15294118 0.14901961 0.16470588 0.21960784]
 [0.14509804 0.16078431 0.18039216 0.16470588 0.19215686]
 [0.18039216 0.16862745 0.16078431 0.18431373 0.19607843]
 [0.18039216 0.19607843 0.19215686 0.19215686 0.18431373]
 [0.20784314 0.20392157 0.23137255 0.22352941 0.17254902]]
```

Gambar 7 Normalisasi Piksel 0-1

Agar proses ekstraksi fitur menggunakan MobileNetV2 dapat berjalan optimal, semua citra akan diubah ukurannya menjadi 244×244 piksel, sesuai dengan ukuran input standar model tersebut [12]. Setelah proses *resize*, dilakukan normalisasi data, yaitu mengubah nilai piksel gambar dari rentang 0 – 255 menjadi rentang 0 – 1. Normalisasi ini bertujuan untuk meningkatkan stabilitas perhitungan dalam jaringan saraf tiruan dan mempercepat proses konvergensi selama pelatihan model.

3.4 Pembagian Dataset

Pembagian dataset dalam penelitian ini bertujuan untuk memastikan bahwa model dapat dilatih secara optimal dan diuji menggunakan data yang tidak pernah digunakan selama pelatihan. Terdapat dua sumber data yang digunakan, yaitu dataset Kaggle dan dataset Mandiri, yang masing-masing dibagi ke dalam tiga skema rasio: 80% data latih, 10% data validasi, dan 10% data uji; kemudian 70% latih, 20% validasi, dan 10% uji; serta skema terakhir yaitu 60% latih, 20% validasi, dan 20% uji. Variasi rasio ini digunakan untuk mengevaluasi pengaruh proporsi data terhadap performa model, serta untuk mengetahui kombinasi yang paling optimal dalam menghasilkan akurasi klasifikasi kerusakan bangunan.

3.5 Ekstraksi Fitur MobileNetV2

Ekstraksi fitur pada penelitian ini dilakukan menggunakan arsitektur MobileNetV2 dengan terlebih dahulu memasukkan citra hasil preprocessing yang telah di-*resize* ke ukuran 244×244 piksel dan dinormalisasi menggunakan fungsi *preprocess_input()* dari TensorFlow/Keras. MobileNetV2 memproses citra melalui lapisan-lapisan konvolusi yang menggunakan teknik *Depthwise Separable Convolution* untuk mengurangi jumlah parameter tanpa mengorbankan kualitas ekstraksi. Fitur penting kemudian diperkuat melalui *Bottleneck Layers* dan diringkas menggunakan *Global Average Pooling*, sehingga dihasilkan vektor satu

dimensi yang merepresentasikan ciri-ciri utama dari citra. Vektor inilah yang selanjutnya digunakan sebagai input bagi model XGBoost untuk proses klasifikasi kerusakan bangunan ke dalam tiga kelas: ringan, sedang, dan berat.

3.6 Klasifikasi XGBoost

Setelah fitur diekstraksi menggunakan MobileNetV2, tahap selanjutnya adalah klasifikasi dengan metode XGBoost, sebuah algoritma ensemble berbasis *decision tree* yang efisien dan akurat, terutama untuk data besar dan kompleks [8]. Label kelas yang awalnya berupa teks dikonversi menjadi numerik menggunakan *Label Encoding* agar dapat diproses oleh XGBoost. Data fitur kemudian dibagi sesuai skema evaluasi (misalnya 80:10:10) menjadi data pelatihan, validasi, dan pengujian. Pada proses pelatihan, XGBoost membangun pohon keputusan secara bertahap dengan teknik *gradient boosting* untuk meminimalkan fungsi loss secara optimal. Selama pelatihan, performa dimonitor menggunakan data validasi dan diterapkan *early stopping* untuk mencegah overfitting [9]. Model hasil pelatihan disimpan dalam format .pkl untuk inferensi selanjutnya. Terakhir, model diuji dengan data uji dan hasil prediksi dibandingkan dengan label asli untuk menghitung metrik evaluasi seperti akurasi, presisi, recall, F1-Score, dan waktu komputasi sebagai dasar penilaian kinerja model.

3.7 Evaluasi dan Pengujian Model

Evaluasi model dimulai dengan menyiapkan dataset uji yang telah dibagi sesuai rasio tertentu dan diproses dengan preprocessing sama seperti data pelatihan. Gambar uji kemudian melewati MobileNetV2 untuk ekstraksi fitur, menghasilkan vektor fitur yang mewakili karakteristik gambar. Vektor ini kemudian diklasifikasikan oleh XGBoost ke dalam tiga kelas kerusakan: ringan, sedang, dan berat. Hasil prediksi dibandingkan dengan label asli untuk menghitung metrik evaluasi seperti akurasi, presisi, recall, dan F1-score, serta dianalisis menggunakan confusion matrix untuk melihat distribusi kesalahan antar kelas. Jika performa model belum memuaskan, dilakukan tuning hyperparameter atau perbaikan dataset untuk meningkatkan akurasi dan kestabilan. Model dengan performa terbaik akhirnya dipilih untuk diterapkan dalam sistem klasifikasi kerusakan bangunan.

3.8 Implementasi GUI

Subbab ini membahas pengembangan Graphical User Interface (GUI) sebagai bagian akhir dari sistem klasifikasi kerusakan bangunan. GUI dirancang untuk memudahkan pengguna dalam berinteraksi dengan sistem tanpa perlu memahami proses teknis di baliknya. Pengguna dapat mengunggah gambar bangunan melalui tombol upload, kemudian gambar diproses melalui MobileNetV2 untuk ekstraksi fitur dan diklasifikasikan oleh model XGBoost. Hasil klasifikasi ditampilkan langsung dalam bentuk teks tingkat kerusakan: “Ringan”, “Sedang”, atau “Berat”. GUI juga menyediakan area pratinjau gambar agar pengguna dapat memastikan gambar yang diproses sudah benar. Antarmuka dibuat sederhana dan ramah pengguna, cocok untuk pengguna non-teknis seperti petugas survei atau masyarakat umum. Implementasi GUI ini menunjukkan bahwa model yang dikembangkan dapat diintegrasikan ke aplikasi nyata dan siap digunakan di lapangan.

4. HASIL DAN PEMBAHASAN

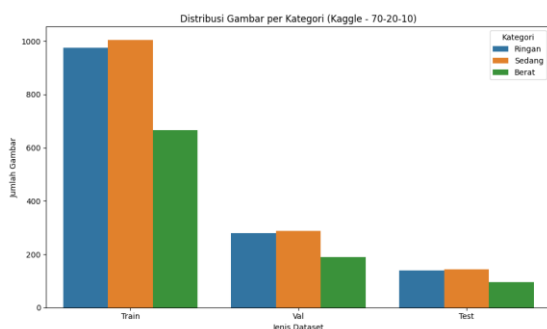
4.1 Metode Pengujian

Pengujian dalam penelitian ini bertujuan untuk mengevaluasi kinerja sistem klasifikasi kerusakan bangunan yang menggunakan ekstraksi fitur MobileNetV2 dan algoritma XGBoost. Pengujian dilakukan untuk menilai kemampuan model dalam mengklasifikasikan tingkat kerusakan bangunan ke dalam tiga kelas—ringan, sedang, dan berat—secara akurat dan efisien. Proses pengujian dilakukan secara eksperimental dengan menggunakan beberapa skema pembagian dataset, yaitu 80:10:10, 70:20:10, dan 60:20:20, pada dua dataset yang berbeda, yakni Kaggle dan Mandiri. Pada tahap ini, citra diubah ukuran dan dinormalisasi sebelum diproses oleh MobileNetV2 untuk ekstraksi fitur, yang kemudian menjadi input bagi model XGBoost. Model dilatih dan diuji menggunakan data fitur tersebut, dengan evaluasi berdasarkan metrik akurasi, presisi, recall, F1-score, serta waktu komputasi. Pengujian ini tidak hanya menilai kualitas klasifikasi, tetapi juga efisiensi waktu proses, sehingga dapat memastikan bahwa metode yang dikembangkan dapat digunakan secara andal dan cepat untuk mendeteksi tingkat kerusakan bangunan secara otomatis.

4.1.1 Pembagian Dataset

Dalam penelitian ini, dataset dibagi

menjadi tiga bagian utama yaitu data train, validation, dan test untuk memastikan pelatihan dan pengujian model berjalan objektif dan sistematis. Data train digunakan untuk melatih model mengenali pola kerusakan bangunan, sedangkan data validation berfungsi memantau performa model selama pelatihan dan mencegah overfitting. Setelah pelatihan, data test digunakan untuk mengevaluasi performa akhir model dengan data yang belum pernah dilihat sebelumnya, sehingga memberikan gambaran kemampuan model terhadap data baru. Pembagian data dilakukan menggunakan skema 80:10:10 yang diterapkan pada dua dataset berbeda, yaitu dataset Kaggle dan dataset mandiri, menghasilkan enam skenario pengujian. Pendekatan ini bertujuan memberikan evaluasi yang komprehensif terhadap kinerja model pada berbagai variasi data.



Gambar 8 Distribusi Data Kaggle 70

Gambar 8 memperlihatkan grafik distribusi jumlah gambar per kategori dalam skema pembagian data 70:20:10 untuk dataset Kaggle. Grafik tersebut menunjukkan bahwa ketiga kelas ringan, sedang, dan berat terdistribusi secara proporsional di masing-masing subset train, validation, dan test. Keseimbangan ini penting untuk menghindari bias model terhadap salah satu kelas. Jumlah data pada subset train tampak jauh lebih banyak dibandingkan validation dan test karena hanya subset ini yang diberi perlakuan augmentasi. Proses augmentasi dilakukan dengan enam teknik transformasi, yakni *horizontal flip*, *vertical flip*, *rotated 45°*, *scaled up*, *shear*, dan *brightness adjustment*. Tujuan augmentasi adalah memperkaya variasi data latih agar model lebih adaptif terhadap bentuk kerusakan yang beragam.

4.1.2 Pengujian Dan Evaluasi Model

Pengujian dilakukan pada data uji yang terpisah dari data latih dan validasi, untuk mengukur kemampuan model dalam mengklasifikasikan citra baru yang belum pernah dilihat. Evaluasi kinerja model menggunakan empat metrik utama, yaitu akurasi, presisi, recall, dan F1-score, yang secara bersama-sama memberikan gambaran lengkap mengenai ketepatan dan sensitivitas model dalam mengenali ketiga kelas kerusakan. Selain itu, waktu komputasi juga dicatat untuk menilai efisiensi proses prediksi. Hasil evaluasi ini menjadi dasar untuk menganalisis keunggulan dan kelemahan model, serta membandingkan performa antar skema pembagian data dan perbedaan antar dataset.

=== Evaluasi Model XGBoost - Kaggle 70:20:10 ===

Akurasi Train : 95.90%
 Akurasi Validation: 91.12%
 Akurasi Test : 93.96%
 Waktu Komputasi : 0.0228 detik

=== Train Classification Report ===

Kelas	Presisi (%)	Recall (%)	F1-Score (%)	Support
0	98.43	93.79	96.05	467
1	95.93	96.77	96.35	682
2	94.30	96.44	95.36	703
Rata-rata	95.94	95.90	95.90	1852

=== Validation Classification Report ===

Kelas	Presisi (%)	Recall (%)	F1-Score (%)	Support
0	95.83	86.47	90.91	133
1	92.35	92.82	92.58	195
2	87.32	92.54	89.86	201
Rata-rata	91.31	91.12	91.13	529

=== Test Classification Report ===

Kelas	Presisi (%)	Recall (%)	F1-Score (%)	Support
0	95.31	91.04	93.13	67
1	95.83	93.88	94.85	98
2	91.43	96.00	93.66	100
Rata-rata	94.04	93.96	93.96	265

Gambar 9 Evaluasi Model Kaggle 70

Gambar 9 memperlihatkan hasil evaluasi model pada skema Kaggle 80:10:10. Model mencapai akurasi 100% pada data latih, serta 91,06% dan 87,13% pada data validasi dan uji. Penurunan ini masih tergolong wajar dan menunjukkan bahwa model tidak overfitting. Performa tinggi pada data uji menunjukkan bahwa fitur dari MobileNetV2 cukup efektif untuk diklasifikasikan oleh XGBoost. Proses inferensi juga efisien, dengan waktu komputasi hanya sekitar 0,0402 detik.

4.2 Hasil Pengujian

Pada bagian ini menyajikan hasil pengujian dari penerapan kombinasi MobileNetV2 dan XGBoost pada dua jenis dataset, yaitu Kaggle dan Mandiri, dengan enam skema pembagian data. Evaluasi

dilakukan menggunakan metrik akurasi, presisi, recall, F1-score, serta waktu komputasi. Hasil juga dilengkapi dengan confusion matrix dan grafik perbandingan performa pada data latih, validasi, dan uji. Melalui analisis ini, dapat diidentifikasi model dan skema pembagian data yang paling optimal untuk klasifikasi kerusakan bangunan.

4.2.1 Akurasi, Presisi, Recall, F1Score

Pengujian model dilakukan untuk mengevaluasi seberapa baik model XGBoost yang telah dilatih mampu mengklasifikasikan citra ke dalam tiga kategori tingkat kerusakan, yaitu ringan, sedang, dan berat. Evaluasi ini menggunakan beberapa metrik utama, yaitu akurasi, presisi, recall, dan F1-score. Masing-masing metrik dihitung berdasarkan prediksi model terhadap data training, validation, dan testing yang telah dibagi sebelumnya dengan skema 80:10:10, 70:20:10, dan 60:20:20

Asal Data	Training : Validation : Testing	Akurasi (%)	Presisi (%)	Recall (%)	F1 - Score (%)	Waktu Komputasi (detik)
Kaggle	80:10:10	89.44	89.65	89.44	89.45	60.53
	70:20:10	94.72	94.80	94.72	94.71	54.91
	60:20:20	83.26	83.38	83.26	83.26	55.54
Mandiri	80:10:10	86.21	86.21	86.21	86.21	53.86
	70:20:10	82.97	82.90	82.97	82.88	48.52
	60:20:20	81.63	81.54	81.63	81.51	41.90

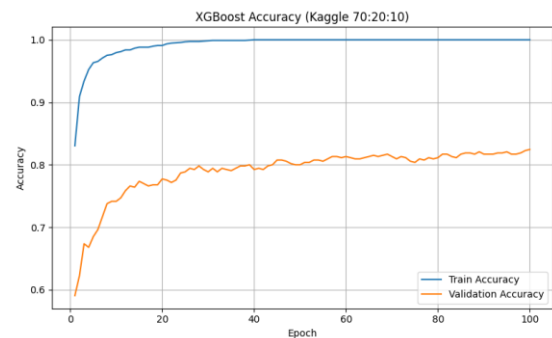
Tabel 1 Hasil Evaluasi Model

Berdasarkan evaluasi terhadap enam skema pembagian data, model hybrid MobileNetV2 dan XGBoost menunjukkan performa yang stabil dan cukup tinggi. Skema 70:20:10 pada dataset Kaggle memberikan hasil terbaik dengan akurasi 94,72% dan waktu komputasi 54,91 detik. Skema lainnya mencatat performa sedikit lebih rendah, terutama pada proporsi data latih yang lebih kecil. Sementara itu, performa pada dataset Mandiri juga konsisten, dengan akurasi tertinggi sebesar 86,21% pada skema 80:10:10. Secara keseluruhan, model tidak menunjukkan gejala overfitting maupun underfitting, dan mampu melakukan klasifikasi dengan baik di berbagai skema dan dataset.

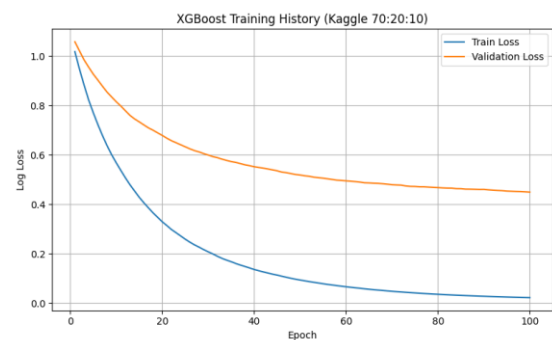
4.2.2 Analisis Berdasarkan Kelas

Analisis ini bertujuan untuk mengevaluasi kinerja model secara spesifik pada masing-masing kelas: ringan, sedang, dan berat. Dengan menghitung metrik seperti presisi, recall, dan F1-score per kelas, dapat diketahui seberapa baik model mengenali tiap

kategori. Metrik tersebut memberikan gambaran lebih mendalam dibanding akurasi keseluruhan, terutama saat distribusi data tidak seimbang. Hasil evaluasi ini membantu mengidentifikasi kelas mana yang paling sulit diprediksi dan menunjukkan area yang perlu diperbaiki untuk meningkatkan performa model secara menyeluruh.



Gambar 10 Accuracy Train dan Val Kaggle 70



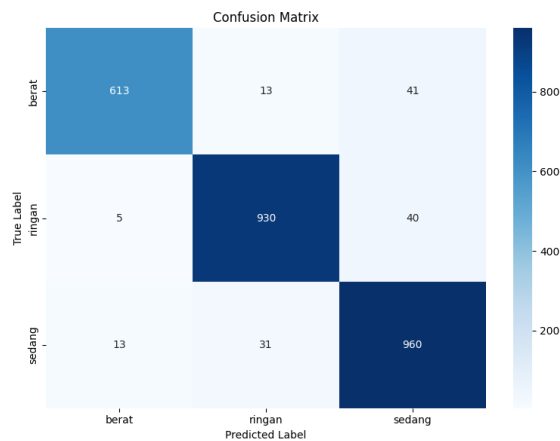
Gambar 11 Loss Train dan Val Kaggle 70

Gambar 10 dan Gambar 11 menunjukkan bahwa akurasi model XGBoost pada data pelatihan meningkat tajam di awal dan cepat mencapai nilai hampir sempurna, sedangkan akurasi validasi meningkat lebih lambat dan stabil di kisaran 80% setelah 40 epoch. Gambar 12 mendukung hasil ini dengan grafik log loss yang terus menurun baik pada data pelatihan maupun validasi, menandakan bahwa model semakin baik dalam memprediksi probabilitas kelas secara akurat dan proses pelatihan berjalan efektif.

4.2.3 Confusion Matrix

Pada bagian ini, evaluasi kinerja model dilakukan menggunakan *confusion matrix* untuk melihat seberapa tepat model dalam mengklasifikasikan data ke tiga kelas kerusakan. Confusion matrix diterapkan pada

masing-masing skema dari dataset Kaggle dan Mandiri, lalu divisualisasikan dalam bentuk *heatmap*. Visualisasi ini memperlihatkan distribusi prediksi benar dan salah, dengan nilai diagonal utama menunjukkan prediksi yang akurat. Melalui pendekatan ini, dapat dilihat pola klasifikasi model secara lebih detail untuk setiap skema.

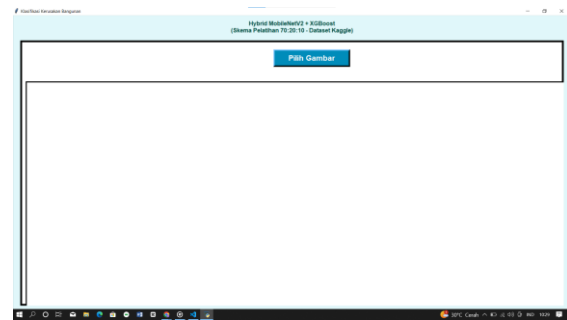


Gambar 12 Confusion Matrix Kaggle 70

Gambar 12 menunjukkan *confusion matrix* pada skema Kaggle 70:20:10. Model menunjukkan performa klasifikasi yang baik, dengan prediksi benar terbanyak pada kelas sedang (960 data), diikuti kelas ringan (930), dan berat (613). Meskipun terdapat beberapa kesalahan, seperti kelas berat yang diprediksi sebagai sedang, jumlahnya relatif kecil. Hasil ini mencerminkan bahwa model mampu mempertahankan akurasi dan generalisasi yang baik, meskipun menggunakan proporsi data latih yang tidak sebesar skema lain.

4.2.4 Tampilan GUI

Subbab ini menjelaskan antarmuka GUI yang dikembangkan untuk mempermudah penggunaan sistem klasifikasi kerusakan bangunan. Melalui tampilan yang sederhana, pengguna dapat mengunggah gambar bangunan, lalu sistem secara otomatis mengekstrak fitur menggunakan MobileNetV2 dan memprediksi tingkat kerusakan dengan XGBoost. Hasil klasifikasi ditampilkan dalam bentuk teks “Ringan”, “Sedang”, dan “Berat” serta disertai pratinjau gambar. GUI ini dirancang agar mudah digunakan oleh siapa saja, termasuk pengguna non-teknis, dan menjadi bukti bahwa model dapat diimplementasikan secara nyata di lapangan.



Gambar 13 Tampilan Awal GUI

Gambar 13 memperlihatkan tampilan awal GUI yang dirancang untuk klasifikasi kerusakan bangunan. Antarmuka ini dibuat sederhana dan mudah diakses, agar dapat digunakan oleh siapa saja, termasuk pengguna non-teknis. Di tengah tampilan terdapat tombol *Upload Gambar* untuk mengunggah citra dari perangkat. Setelah gambar dipilih, sistem secara otomatis menjalankan proses ekstraksi fitur dengan MobileNetV2 dan klasifikasi menggunakan XGBoost, lalu menampilkan hasil prediksi secara langsung.



Gambar 14 Tampilan GUI Hasil Prediksi

Gambar 14 menunjukkan tampilan GUI setelah pengguna mengunggah gambar dan sistem selesai memprediksi. Antarmuka menampilkan hasil klasifikasi yang diperoleh dari ekstraksi fitur menggunakan MobileNetV2 dan klasifikasi dengan XGBoost. Hasil prediksi berupa tingkat kerusakan bangunan, yaitu Ringan, Sedang, atau Berat, langsung ditampilkan secara otomatis setelah proses selesai.

5. KESIMPULAN

Penelitian ini berhasil mengembangkan metode hybrid dengan menggabungkan MobileNetV2 sebagai ekstraktor fitur dan XGBoost sebagai klasifikator untuk mengidentifikasi tingkat kerusakan bangunan

ke dalam tiga kelas: ringan, sedang, dan berat. Evaluasi dilakukan menggunakan dua dataset berbeda, yaitu Kaggle dan Mandiri, dengan tiga skema pembagian data yang variatif. Hasil pengujian menunjukkan performa terbaik diperoleh pada dataset Kaggle dengan skema 70:20:10, yang mencapai akurasi dan metrik evaluasi lain di atas 94%, serta waktu komputasi yang efisien sekitar 54,91 detik. Pada dataset Mandiri, skema 80:10:10 memberikan hasil terbaik dengan akurasi dan metrik lainnya sekitar 86%, meskipun masih lebih rendah dibandingkan Kaggle, namun tetap menunjukkan kestabilan performa.

Analisis pelatihan juga tidak menunjukkan tanda-tanda overfitting maupun underfitting, yang mengindikasikan model mampu belajar secara efektif dan tetap memiliki kemampuan generalisasi yang baik pada data baru. Secara keseluruhan, kombinasi MobileNetV2 dan XGBoost terbukti efektif dan efisien dalam klasifikasi kerusakan bangunan, memberikan hasil yang akurat dan konsisten. Penelitian ini memenuhi tujuan utamanya dengan menunjukkan bahwa metode hybrid ini adaptif dan dapat diandalkan pada berbagai kondisi dataset, sehingga berpotensi menjadi solusi praktis dalam mempercepat proses identifikasi tingkat kerusakan bangunan di lapangan.

UCAPAN TERIMA KASIH

Penulis menyampaikan rasa terima kasih yang sebesar-besarnya kepada semua pihak yang telah memberikan dukungan, bimbingan, dan bantuan selama proses penelitian ini berlangsung. Ucapan terima kasih khusus disampaikan kepada dosen pembimbing yang telah memberikan arahan dan motivasi, sehingga penelitian ini dapat diselesaikan dengan baik. Selain itu, penulis juga berterima kasih kepada keluarga, teman-teman, dan rekan-rekan yang telah memberikan dukungan moral dan semangat. Tidak lupa, penulis mengapresiasi semua pihak yang terkait, baik secara langsung maupun tidak langsung, yang telah membantu dalam pengumpulan data, penyediaan fasilitas, serta dukungan teknis lainnya. Semoga hasil penelitian ini dapat memberikan manfaat dan kontribusi positif bagi pengembangan ilmu pengetahuan dan teknologi.

DAFTAR PUSTAKA

- [1] S., Rimba K., et al. "Identifikasi Dampak dan Tingkat Serangan Rayap terhadap Bangunan di Kabupaten Kuantan Singingi." *Jurnal Online Mahasiswa Fakultas Pertanian Universitas Riau*, vol. 2, no. 2, Oct. 2015, pp. 1-9.
- [2] Dardiri, A. (2013). Analisis Pola, Jenis, dan Penyebab Kerusakan Bangunan Gedung Sekolah Dasar. *Teknologi dan Kejuruan*, 35(1).
- [3] N. Hikmatia A.E and M. Ihsan Zul, "Aplikasi Penerjemah Bahasa Isyarat Indonesia menjadi Suara berbasis Android menggunakan Tensorflow," *J. Komput. Terap.*, vol. 7, no. Vol. 7 No. 1 (2021), pp. 74 – 83, 2021, doi: 10.35143/jkt.v7i1.4629.
- [4] Yulianti, S. E. H., Soesanto, O., & Sukmawaty, Y. (2022). Penerapan Metode Extreme Gradient Boosting (XGBOOST) pada Klasifikasi Nasabah Kartu Kredit. *Journal of Mathematics: Theory and Applications*, 21-26.
- [5] Adhinata, F. D., Tanjung, N. A. F., Widayat, W., Pasfica, G. R., & Satura, F. R. (2021). Comparative study of VGG16 and MobileNetv2 for masked face recognition. *J. Ilm. Tek. Elektro Komput. dan Inform*, 7(2), 230-237.
- [6] Ae, N. H., & Zul, M. I. (2021). Aplikasi Penerjemah Bahasa Isyarat Indonesia menjadi Suara berbasis Android menggunakan Tensorflow. *Jurnal Komputer Terapan*, 7(1), 74-83.
- [7] Karo, I. M. K. (2020). Implementasi metode XGBoost dan feature important untuk klasifikasi pada kebakaran hutan dan lahan. *Journal of Software Engineering, Information and Communication Technology (SEICT)*, 1(1), 11-18.
- [8] Yulianti, S. E. H., Soesanto, O., & Sukmawaty, Y. (2022). Penerapan Metode Extreme Gradient Boosting (XGBOOST) pada Klasifikasi Nasabah Kartu Kredit. *Journal of Mathematics: Theory and Applications*, 21-26.
- [9] Septiantio, A. S. (2023). Klasifikasi Penyakit Paru-Paru Berdasarkan Citra X-Ray Dengan Metode Extreme Gradient Boost (XGBOOST) (Doctoral dissertation, Institut Teknologi Sepuluh Nopember).
- [10] Kusumanto, R. D., & Tompunu, A. N. (2011, April). pengolahan citra digital untuk mendeteksi obyek menggunakan pengolahan warna model normalisasi RGB. In *Seminar Nasional Teknologi Informasi & Komunikasi Terapan* (Vol. 2011, pp. 1-7).
- [11] Santoso, A., & Gunawan Ariyanto, S. T. (2018). Implementasi deep learning berbasis keras untuk pengenalan wajah (Doctoral

- dissertation, Universitas Muhammadiyah Surakarta).
- [12] Marpaung, F., Khairina, N., Muliono, R., Muhathir, M., & Susilawati, S. (2024). Klasifikasi Daun Teh Siap Panen Menggunakan Convolutional Neural Network Arsitektur Mobilenetv2. *Jurnal Teknoinfo*, 18(1), 215-225.
- [13] Tripathi, A., Singh, T., Nair, R. R., & Duraisamy, P. (2024). Improving Early Detection and Classification of Lung Diseases with Innovative MobileNetV2 Framework. *IEEE Access*.
- [14] Ali, Z. A., Abduljabbar, Z. H., Tahir, H. A., Sallow, A. B., & Almufti, S. M. (2023). eXtreme gradient boosting algorithm with machine learning: A review. *Academic Journal of Nawroz University*, 12(2), 320-334.