

ANALISIS KOMPARATIF KINERJA RESNET-50, EFFICIENTNET-B0, DAN MOBILENETV3-LARGE DALAM KLASIFIKASI KERUSAKAN DINDING BANGUNAN BERBASIS CITRA DIGITAL

Nurhidayah Ramadhani, Kristian Aji Suseno, Immanuel Gracio Moningkey,
Noveria Mumek, Pujo Hari Saputro, Sherwin Aldo Sompie

Teknik Informatika, Universitas Sam Ratulangi
Jalan Kampus Bahu, Kecamatan Malalayang, Kota Manado, Indonesia
nurhidayahramadhani026@student.unsrat.ac.id

ABSTRAK

Indonesia merupakan wilayah rawan gempa, seperti kejadian gempa magnitudo 7,6 di perairan Sulawesi Utara dan Maluku Utara pada April 2026 yang mengakibatkan kerusakan masif pada struktur dinding bangunan. Inspeksi bangunan pasca-bencana secara manual membutuhkan waktu lama, bersifat subjektif, dan berisiko tinggi. Meskipun teknologi *deep learning* dapat mengotomatisasi proses ini, setiap arsitektur jaringan memiliki kompromi (*trade-off*) yang berbeda antara akurasi, ukuran model, dan kecepatan inferensi. Penelitian ini bertujuan melakukan analisis komparasi kinerja tiga arsitektur *deep learning*, yaitu ResNet-50, EfficientNet-B0, dan MobileNetV3-Large, dalam mengklasifikasikan enam kategori kerusakan dinding bangunan berbasis citra digital menggunakan pendekatan *transfer learning*. Dataset yang digunakan berjumlah 30.000 citra yang menggabungkan data lapangan wilayah Manado dan dataset publik. Hasil pengujian pada *test set* independen (3.000 citra) menunjukkan bahwa ResNet-50 menghasilkan akurasi tertinggi sebesar 99,40%, disusul oleh MobileNetV3-Large sebesar 99,37%, sementara EfficientNet-B0 menghasilkan performa terendah sebesar 95,77%. Meskipun ResNet-50 memuncaki metrik akurasi, model tersebut berukuran file 207,18 MB dengan kecepatan inferensi 336,14 ms/citra. Sebaliknya, MobileNetV3-Large hanya berukuran 27,34 MB dengan kecepatan inferensi *real-time* sebesar 49,29 ms/citra. Berdasarkan keseimbangan antara akurasi tinggi dan efisiensi komputasi, MobileNetV3-Large direkomendasikan sebagai arsitektur terbaik untuk diimplementasikan pada perangkat *mobile* guna mendukung sistem inspeksi kerusakan bangunan otomatis pasca-bencana secara cepat di lapangan.

Kata kunci : *Deep learning, Klasifikasi Kerusakan Dinding, MobileNetV3-Large, ResNet-50, EfficientNet-B0, Citra Digital.*

1. PENDAHULUAN

Bangunan merupakan infrastruktur vital yang secara langsung menopang keselamatan aktivitas manusia sehari-hari. Namun, kerusakan elemen struktural bangunan khususnya dinding akibat faktor usia bangunan maupun bencana alam menjadi permasalahan masif di Indonesia. Berdasarkan katalog gempa bumi dan tsunami dari Badan Meteorologi, Klimatologi, dan Geofisika (BMKG), Indonesia berada di wilayah tektonik aktif yang rentan terhadap kegagalan struktural bangunan akibat guncangan tanah [1].

Kerusakan yang tidak terdeteksi secara dini, seperti retakan, pertumbuhan lumut, pengelupasan cat, hingga noda dan *spalling*, merupakan indikator awal penurunan kondisi bangunan yang dapat berkembang menjadi kegagalan struktural fatal yang mengancam keselamatan penghuninya [2] [3].

Urgensi penanganan kerusakan bangunan terbukti dalam peristiwa gempa berkekuatan magnitudo 7,6 di wilayah Maluku Utara dan Sulawesi Utara pada 2 April 2026 yang mengakibatkan kerusakan serius pada puluhan bangunan hunian dan fasilitas umum di Kota Manado, Bitung, dan sekitarnya [4].

Dalam situasi pasca-bencana, proses penilaian kondisi keamanan bangunan harus dilakukan sebelum

bangunan kembali digunakan. Metode inspeksi konvensional oleh ahli terlatih masih menghadapi tantangan besar berupa keterbatasan personel, subjektivitas penilaian, serta risiko keselamatan surveyor di area terdampak [5] [6].

Lebih jauh, adopsi teknologi kecerdasan buatan dalam inspeksi manual di Indonesia menghadapi hambatan seperti minimnya ketersediaan dataset citra kerusakan bangunan berlabel lokal, serta belum adanya tolak ukur yang membandingkan kinerja arsitektur *deep learning* secara komprehensif dalam penanganan pasca-bencana nasional. Kedua faktor inilah yang menyebabkan solusi *Convolutional Neural Network* (CNN) belum diadopsi secara luas oleh praktisi maupun instansi penanggulangan bencana di lapangan.

Perkembangan teknologi *deep learning*, khususnya CNN menawarkan solusi otomatisasi inspeksi bangunan yang objektif dan efisien, serta pemanfaatan teknik *transfer learning* terbukti efektif dalam mengenali pola kerusakan infrastruktur dengan data terbatas melalui adaptasi model *pre-trained* [9] [10].

Namun, penerapan CNN pada skenario tanggap darurat pasca-bencana sering kali dihadapkan pada dilema antara kebutuhan akurasi prediksi yang tinggi dan keterbatasan sumber daya komputasi perangkat

mobile di lapangan. Oleh karena itu, analisis komparatif kinerja arsitektur menjadi sangat krusial untuk mengevaluasi trade-off tersebut. Penelitian ini memilih tiga arsitektur yang merepresentasikan spektrum karakteristik teknis yang berbeda dan saling melengkapi: ResNet50 menerapkan *residual learning* untuk mengatasi degradasi pada jaringan dalam yang unggul dalam kapasitas representasi fitur dan akurasi [9]; EfficientNet-B0 mengoptimalkan keseimbangan antara kedalaman, lebar, dan resolusi jaringan melalui *compound scaling* yang menawarkan titik tengah antara akurasi dan efisiensi [10]; serta MobileNetV3-Large untuk efisiensi tinggi pada perangkat dengan keterbatasan sumber daya melalui kombinasi *automated search* dan *network redesign*, menjadi kandidat utama untuk *deployment* pada perangkat *mobile* [11].

Pemilihan ketiga arsitektur ini alih-alih VGG16 atau InceptionV3, didasarkan pada relevansinya dengan skenario *deployment* di lapangan pasca-bencana yang menuntut keseimbangan antara kinerja dan efisiensi komputasi [12].

Dalam penelitian ini, kerusakan dinding diklasifikasikan ke dalam enam kategori berdasarkan prevalensinya dalam inspeksi bangunan pasca-bencana di Indonesia, yaitu: dinding bagus, retak, rusak, cat terkelupas, noda, dan berlumut/berjamur. Keenam kategori ini dipilih karena merepresentasikan tingkat urgensi penanganan yang berbeda dalam asesmen keselamatan bangunan: retak dan rusak mengindikasikan potensi kegagalan struktural yang memerlukan tindakan segera, sementara cat terkelupas, noda, dan pertumbuhan lumut/jamur merupakan indikator degradasi non-struktural yang tetap penting dipantau karena dapat menjadi gejala awal masuknya kelembapan atau kerusakan yang lebih dalam pada dinding. Dengan mengombinasikan kategori struktural dan non-struktural dalam satu skema klasifikasi, sistem yang dibangun mampu mendukung proses triase kondisi bangunan secara lebih menyeluruh dan sesuai dengan kebutuhan riil inspeksi lapangan pasca-bencana.

Sejumlah penelitian terdahulu telah mengeksplorasi perbandingan performa antar arsitektur dalam berbagai domain. Model *hybrid* berbasis ResNet-50 telah diusulkan untuk deteksi kerusakan struktural berbasis citra [13].

Efektivitas CNN berbasis *transfer learning* menggunakan MobileNet juga telah didemonstrasikan untuk asesmen cepat kerusakan bangunan pasca-gempa [5].

Selain itu, pengembangan CNN juga dilakukan untuk klasifikasi *multi-level* kerusakan bangunan menggunakan citra udara pasca-bencana [14].

Di tingkat nasional, penelitian terbaru mulai membandingkan performa arsitektur-arsitektur ini seperti pada klasifikasi kerusakan jalan menggunakan

ResNet50 dan EfficientNet-B0 [17] [18], komparasi MobileNetV3-Large dan EfficientNet-B0 [19] [20].

Sementara itu, di domain medis ResNet50 dan EfficientNet-B0 juga terbukti menghasilkan performa yang berbeda secara signifikan bergantung pada karakteristik *dataset* yang digunakan [19], serta deteksi kerusakan retak bangunan berbasis citra digital menggunakan MobileNetV2 dan augmentasi data [3].

MobileNetV3-Large mulai banyak dikaji untuk deteksi retak pada perangkat *edge computing* karena keunggulannya dalam efisiensi komputasi [10] [22], sementara penanganan *dataset* tidak seimbang melalui strategi augmentasi terbimbing proporsional dan teknik *early stopping* terbukti krusial dalam skenario klasifikasi multi-kelas di lapangan [21].

Namun, hingga saat ini belum ada studi yang secara spesifik membandingkan ketiga arsitektur tersebut menggunakan *dataset* hibrida yang memadukan citra lapangan wilayah terdampak bencana di Sulawesi Utara dengan *dataset* publik *Building Dataset* pada domain klasifikasi enam kategori kerusakan dinding secara bersamaan dan kemudian mengevaluasi performa arsitektur-arsitektur tersebut dalam skenario data tidak seimbang yang merupakan kondisi riil di lapangan pasca-bencana di Indonesia Timur.

Berdasarkan latar belakang tersebut, penelitian ini bertujuan mengkomparasi kinerja arsitektur ResNet50, EfficientNet-B0, dan MobileNetV3-Large dalam mengklasifikasikan enam kategori kerusakan dinding bangunan berbasis citra digital dengan *dataset* berjumlah 30.000 citra setelah proses augmentasi, mengintegrasikan 6.000 citra bersih hasil penggabungan data lapangan wilayah terdampak di Manado dengan *dataset* publik, melalui serangkaian tahap pra-proses ketat meliputi *deduplication* berbasis *hashing* MD5, peningkatan kualitas citra selektif menggunakan *Contrast Limited Adaptive Histogram Equalization* (CLAHE) pada kanal *Lightness* (L) dalam ruang warna LAB, serta enam *pipeline* augmentasi untuk memastikan keragaman dan ketangguhan model [23].

Optimasi dilakukan melalui strategi *transfer learning* dua fase disertai teknik *early stopping* untuk mencegah *catastrophic forgetting* [23].

Penelitian ini memberikan tiga kontribusi utama: (1) pembangunan *dataset* kerusakan dinding berbasis citra lapangan wilayah Sulawesi Utara yang dapat menjadi referensi lokal untuk penelitian sejenis; (2) *benchmark* komparatif tiga arsitektur CNN terkemuka pada domain inspeksi bangunan dengan enam kelas kerusakan dalam konteks pasca-bencana; dan (3) rekomendasi arsitektur berbasis bukti untuk implementasi sistem inspeksi bangunan mandiri pada perangkat *mobile* guna mendukung percepatan asesmen keselamatan masyarakat pasca-bencana di Indonesia Timur.

2. TINJAUAN PUSTAKA

2.1. Kerusakan Dinding dan Inspeksi Berbasis Citra

Kerusakan permukaan dinding merupakan indikator awal penurunan integritas struktural bangunan [2].

Ekstraksi fitur visual dari citra digital menjadi fondasi utama sistem asesmen keselamatan yang objektif guna mereduksi keterbatasan metode inspeksi konvensional, seperti subjektivitas penilaian, keterbatasan jumlah surveyor, serta risiko keselamatan kerja di area pasca-bencana [5] [6] [16].

2.2. Praproses dan Penanganan Class Imbalance

Kualitas citra masukan memengaruhi kemampuan *Convolutional Neural Network* (CNN) dalam mengekstraksi fitur diskriminatif tekstur halus kerusakan dinding [22].

Penerapan CLAHE secara selektif pada kanal L ruang warna LAB terbukti efektif meningkatkan ketajaman detail lokal tanpa mendistorsi informasi warna pada kanal A dan B [20].

Di sisi lain, distribusi data riil yang tidak merata memicu masalah *class imbalance* yang membuat model CNN bias terhadap kelas mayoritas [21].

Untuk mengatasinya, strategi augmentasi data terbimbing dengan enam *pipeline* transformasi kombinatorial (geometris, kecerahan, sistematis *noise* kamera, *zoom-crop*, penajaman tepi, dan *color jitter*) diaplikasikan secara bergantian hingga mencapai distribusi berimbang 5.000 citra per kelas [23].

Penerapan teknik augmentasi kombinatorial secara dinamis ini terbukti mampu meningkatkan akurasi klasifikasi secara signifikan dan memperkuat kemampuan generalisasi model pada variasi kondisi lapangan [3].

2.3. Transfer learning dan Pelatihan Dua Fase

Transfer learning memanfaatkan representasi fitur hierarkis yang diperoleh dari arsitektur *pre-trained* pada *dataset* berskala besar (*ImageNet*) untuk menyelesaikan masalah klasifikasi pada domain target dengan data terbatas. Implementasi dalam penelitian ini mengadopsi strategi dua fase yang secara konsisten terbukti mengungguli metode ekstraksi fitur tunggal [5].

- a. Fase 1 (*Feature Extraction*): Seluruh lapisan *convolutional base* dibekukan (*base.trainable = False*) untuk mengoptimalkan bobot *classifier head* baru dalam memetakan fitur *ImageNet* ke enam kelas target kerusakan menggunakan *optimizer* Adam dengan *learning rate* 0,001 selama 75 epoch.
- b. Fase 2 (*Fine-Tuning*): Sebanyak 30 lapisan terakhir dari *base model* dibuka kembali (*base.trainable = True*) dengan penurunan *learning rate* secara drastis menjadi 0,0001 guna memitigasi risiko *catastrophic forgetting* [23].

2.4. Arsitektur CNN Pembanding

Eksperimen ini mengevaluasi tiga arsitektur CNN dengan karakteristik topologi yang berbeda untuk menemukan performa optimal:

- a. ResNet50: Mengintegrasikan mekanisme *residual/skip connection* yang membentuk jalur pintas (*shortcut path*) untuk menghubungkan keluaran suatu blok lapisan langsung ke lapisan berikutnya melewati transformasi fungsi residu $F(x) = H(x) - x$ [9].
Jalur ini mengalirkan gradien langsung ke lapisan awal saat *backpropagation*, secara efektif mengatasi masalah *vanishing gradient* pada jaringan dalam [9].
ResNet50 terbukti andal dalam ekstraksi fitur klasifikasi retak beton dan sering dikombinasikan dengan mekanisme perhatian untuk memfokuskan model pada area informatif kerusakan [10] [15].
- b. EfficientNet-B0: Menggunakan pendekatan *compound scaling* yang mengoptimalkan dimensi kedalaman, lebar, dan resolusi input jaringan secara simultan melalui koefisien tunggal berbasis *Neural Architecture Search* (NAS) [10].
Pendekatan ini menghasilkan model yang efisien secara parameter namun tetap mempertahankan akurasi tinggi, menjadikannya ideal untuk rasio pembagian data seimbang [20] [21].
- c. MobileNetV3-Large: Dikembangkan lewat kombinasi *hardware-aware NAS*, algoritma NetAdapt, serta penambahan blok *squeeze-and-excitation* dan fungsi aktivasi *hard-swish* yang efisien dioperasikan tanpa membebani perangkat keras [11].
Topologi ini menghasilkan ukuran parameter sangat ringkas yang kompetitif untuk klasifikasi retak infrastruktur, baik pada citra resolusi rendah maupun tinggi, dengan waktu pelatihan paling singkat sehingga ideal untuk integrasi gawai *mobile* maupun UAV secara *real-time* [6] [13] [14] [22].

2.5. Metrik Evaluasi Klasifikasi dan Komputasi

Evaluasi komprehensif dilakukan menggunakan multi-metrik yang saling melengkapi [19].

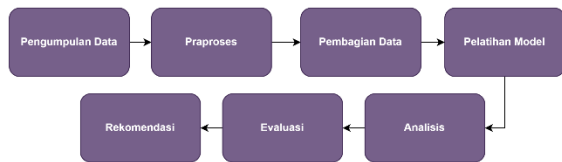
Performa klasifikasi diukur melalui metrik *accuracy* untuk proporsi prediksi benar pada *dataset* seimbang, *precision* untuk keandalan prediksi positif, *recall* untuk sensitivitas penjarangan sampel kerusakan, dan *F1-Score* sebagai rata-rata harmonik penyeimbang [15].

Kualitas kesepakatan prediksi sistem diuji secara ketat menggunakan *Cohen's Kappa Score* berbasis skala Landis & Koch untuk mengeliminasi faktor kebetulan (*chance agreement*) [2].

Metrik *Area Under the ROC Curve* (AUC) dengan pendekatan *One-vs-Rest* (OvR) digunakan untuk mengukur kemampuan diskriminasi biner model pada tiap kelas individu [2].

Sementara itu, kesiapan implementasi model pada perangkat *mobile* (*edge deployment*) diuji melalui tiga parameter efisiensi komputasi fisik: jumlah parameter model aktif (juta), ukuran file model tersimpan (MB), dan kecepatan inferensi rata-rata (*milisecond per image*, ms/img) [6], [8], [12].

3. METODE PENELITIAN



Gambar 1. Diagram alur tahapan penelitian

3.1. Pengumpulan Data

Penelitian ini diawali dengan melakukan pengumpulan *dataset* yang bersumber dari dua entitas berbeda. Kumpulan data pertama merupakan *dataset* publik daring (*online*) yang diunduh melalui repositori *Building Dataset* yang terdiri atas tujuh folder asli [3] [9], sedangkan kumpulan data kedua merupakan lapangan (*real*) yang diakuisisi mandiri secara langsung menggunakan kamera gawai pada area terdampak gempa bumi Sulawesi Utara di Kota Manado [1] [4].

Kedua kumpulan *dataset* dari sumber *online* dan *real* tersebut selanjutnya digabungkan menjadi satu repositori sebelum memasuki tahapan manipulasi citra digital.

3.2. Praproses Citra Digital

Pada tahap kedua, pengolahan data berfokus pada pemetaan folder, deduplikasi, penyesuaian ukuran, perbaikan kontras lokal citra, dan augmentasi data. Tahap awalnya adalah memetakan tujuh folder asal dari *dataset* publik menjadi enam kelas target penelitian [3].

Pemetaan ini menggabungkan folder *major_crack* dan *minor_crack* ke dalam kelas *dinding_retak*. Penyesuaian nama juga diterapkan pada kelas lain: *normal* menjadi *dinding_bagus*, *algae* menjadi *dinding_berlumut_berjamur*, *peeling* menjadi *dinding_cat_terkelupas*, *spalling* menjadi *dinding_rusak*, dan *stain* menjadi *dinding_noda*. Pengumpulan data mentah awal menghasilkan 6.305 gambar. Selama proses penyalinan, data publik ditambahkan prefiks *online* dan data lapangan ditambahkan prefiks *real*.

Proses pembersihan data dilakukan menggunakan metode *deduplication* berbasis *hashing* biner MD5, yang berhasil menghapus 318 citra duplikat sehingga tersisa 5.987 citra bersih. Seluruh citra kemudian diubah ukurannya menjadi 224×224 piksel menggunakan interpolasi Lanczos4 (*cv2.INTER_LANCZOS4*).

Peningkatan kualitas citra diterapkan secara selektif menggunakan CLAHE pada ruang warna LAB (kanal L) dengan parameter *clip limit* = 2,0 dan *tile*

grid size = (8,8), khusus untuk citra yang memenuhi syarat: *mean brightness* < 100 atau standar deviasi kontras < 40 [9].

Hasil citra disimpan dalam format JPEG dengan kualitas kompresi 95%. Tahap praproses diakhiri dengan ekspansi data melalui enam *pipeline* augmentasi berbeda yang diaplikasikan secara bergantian (*round-robin*) menggunakan pustaka *albumentations* hingga setiap kelas mencapai target seimbang sempurna tepat 5.000 citra (total akhir *dataset* 30.000 citra) [3] [23].

Struktur *pipeline* tersebut meliputi: (1) *Pipeline 1* (*Flip & Rotasi Ringan*): *Horizontal flip* ($p=0,5$), *vertical flip* ($p=0,3$), dan rotasi acak $\pm 10^\circ$ berbasis *border mode reflect*. (2) *Pipeline 2* (*Kecerahan & Warna*): *Horizontal flip*, penyesuaian kecerahan dan kontras acak ($\pm 0,15$), pergeseran nilai HSV, dan penyesuaian *gamma* dalam rentang (90, 110). (3) *Pipeline 3* (*Noise Minimal*): Injeksi *Gaussian noise* sangat ringan (*std_dev* = 0,01 - 0,03, $p=0,3$) dan *Gaussian blur* minimal dengan *limit* (3,3) dan $p=0,25$. (4) *Pipeline 4* (*Zoom & Crop*): Operasi *RandomResizedCrop* dengan skala (0,9-1,0) dan rasio aspek (0,95-1,05) dikombinasikan dengan modifikasi kecerahan ringan. (5) *Pipeline 5* (*CLAHE & Penajaman*): Aplikasi CLAHE (*clip limit* = 2,0) dan *Sharpen* dengan intensitas ringan ($\alpha = 0,05 - 0,2$). (6) *Pipeline 6* (*Color Jitter Ringan*): Kombinasi *horizontal flip*, *vertical flip* ($p=0,2$), dan *ColorJitter* mikro (parameter *brightness* = *contrast* = *saturation* = 0,1; *hue* = 0,03) [3] [23].

3.3. Pembagian Data

Dataset hasil augmentasi yang seimbang sempurna (30.000 citra) didistribusikan alokasinya menggunakan metode pembagian berjenjang (*stratified split*). Skema pembagian diatur secara deterministik dengan proporsi rasio 70/20/10, di mana sebesar 70% (21.000 citra) dialokasikan sebagai *subset* data latih (*training set*). Sisa data sebesar 30% (9.000 citra) dibagi kembali secara internal dengan porsi 20% dari total *dataset* sebagai data validasi (*validation set* 6.000 citra) dan 10% sisanya sebagai data uji independen (*test set* 3.000 citra, yaitu 500 citra per kelas secara konsisten). Pengondisian pembagian data dikunci menggunakan parameter *seed*=42 guna menjamin reproduktibilitas eksperimen. Seluruh *dataset* dimuat menggunakan fungsi *tf.keras.utils.image_dataset_from_directory* pada ukuran *batch size* = 32 dengan dimensi spasial 224 × 224 piksel. Selama proses pelatihan, khusus untuk *subset training set* diberikan tambahan fungsi augmentasi dinamis secara *online* berupa *horizontal flip* acak, rotasi acak $\pm 10^\circ$ ($\pm 36^\circ$), dan *zoom* acak $\pm 10\%$ [21].

Aliran seluruh *pipeline* data dioptimasi menggunakan instruksi *tf.data.AUTOTUNE* dan metode *prefetch()* demi menghilangkan risiko hambatan memori (*bottleneck*) komputasi GPU.

3.4. Pelatihan Model

Tahap keempat mengeksekusi proses komputasi paralel menggunakan tiga arsitektur model yakni ResNet50 [9], EfficientNet-B0 [10], dan MobileNetV3-Large [11] yang memanfaatkan bobot *transfer* awal dari *ImageNet* [5] [9] [10] [15]. Guna menjamin asas keadilan komparasi (*fair comparison*), bagian lapisan pengklasifikasi asli bawaan (*top layer*) dari ketiga model dibuang secara permanen dan digantikan oleh susunan arsitektur *classifier head* baru yang seragam [17] [19] [21].

Struktur *classifier head* artifisial tersebut disusun secara linear berturut-turut meliputi:

- Lapisan *GlobalAveragePooling2D* untuk meratakan peta fitur menjadi vektor satu dimensi secara spasial sekaligus memitigasi risiko *overfitting*,
- Lapisan *BatchNormalization* untuk menstabilkan serta mempercepat konvergensi laju pelatihan,
- Lapisan *Dense* dengan 256 unit *node* disertai fungsi aktivasi *Rectified Linear Unit* (ReLU),
- Lapisan *Dropout* dengan parameter *rate* = 0,5 sebagai fungsi regularisasi neuron,
- Lapisan *Dense* sekunder dengan 128 unit *node* diiringi aktivasi ReLU,
- Lapisan *Dropout* sekunder dengan parameter *rate* = 0,3,
- Lapisan *Dense* akhir dengan 6 unit *node* yang dikombinasikan dengan fungsi aktivasi *Softmax* untuk mengeluarkan nilai distribusi probabilitas prediksi klasifikasi enam kelas kerusakan.

Fungsi kerugian (*loss function*) yang diimplementasikan adalah *categorical cross-entropy* dikombinasikan dengan pemantauan metrik *accuracy*, *precision*, dan *recall*.

Siklus pelatihan berjalan melalui skema *Two-Phase Transfer learning* untuk memaksimalkan adaptasi fitur sekaligus menghalau bahaya gejala *catastrophic forgetting* [23].

Fase 1 (*Feature Extraction*) berjalan penuh selama 75 epoch dengan status seluruh lapisan dasar dibekukan (*base.trainable=False*) menggunakan *optimizer* Adam pada nilai laju pembelajaran awal LR = 0,001 [5].

Memasuki fase 2 (*Fine-Tuning* dengan batas maksimum 225 epoch), bobot performa terbaik dari fase 1 dimuat kembali, lalu status lapisan dasar dibuka secara selektif (*base.trainable=True*) khusus untuk 30 lapisan konvolusi terdalam saja [5] [10].

Nilai laju pembelajaran diturunkan secara halus menjadi LR = 0,0001 [5] [9].

Pelatihan dikendalikan oleh empat *callback* paralel: *ModelCheckpoint* (monitor *val_accuracy*), *EarlyStopping* (*patience*=20, *restore_best_weights=True*), *ReduceLROnPlateau* (*patience*=10, *factor*=0,2, monitor *val_loss*), dan *CSVLogger* [23].

Fase 2 mencapai konvergensi dan dihentikan *early stopping* pada epoch ke-179 (ResNet50), ke-160

(EfficientNet-B0), dan ke-199 (MobileNetV3-Large) [17] [19].

Seluruh proses pelatihan dan pengujian model dijalankan pada platform Google Colaboratory (*free tier*) dengan spesifikasi perangkat keras berupa GPU NVIDIA Tesla T4 (16 GB VRAM), RAM 12,7 GB, dan prosesor Intel Xeon (virtual CPU, dialokasikan dinamis oleh Google), dengan lingkungan pengembangan TensorFlow/Keras versi 2.15 pada sistem operasi Ubuntu (Linux, berbasis cloud). Spesifikasi ini memengaruhi hasil pengukuran kecepatan inferensi (ms/citra) yang dilaporkan pada Tabel 3 sehingga perlu dicantumkan untuk menjamin reproduktibilitas eksperimen.

3.5. Evaluasi Model

Model yang telah menyelesaikan seluruh siklus pelatihan dimuat kembali berkas bobot optimalnya dari file *fase2_best.weights.h5*. Pengujian performa dilakukan secara objektif memanfaatkan subset data uji (*unseen test set*) berjumlah total 3.000 citra yang tersimpan terisolasi di dalam direktori *dataset_test/* (500 citra seimbang sempurna per kelas). Evaluasi dijalankan menggunakan pendekatan *mini-batch* berukuran kecil (*batch size* = 8 citra) demi menjamin efisiensi alokasi memori RAM. Setiap model mengeksekusi fungsi normalisasi piksel bawaan arsitektur orisinalnya masing-masing, meliputi *resnet50.preprocess_input* [9], *efficientnet.preprocess_input* [10], dan *mobilenet_v3.preprocess_input* [11] untuk menjamin kesesuaian rentang skala *ImageNet*. Kualitas keandalan model dinilai dari dua dimensi pengukuran. Dimensi pertama fokus pada performa statistik klasifikasi mencakup nilai *accuracy*, *precision* (*weighted*), *recall* (*weighted*), *F1-Score* (*weighted*), dan *Cohen's Kappa Score* berbasis standarisasi batasan skala Landis & Koch, serta visualisasi kurva *Area Under the ROC Curve* (AUC) dengan pendekatan metode *One-vs-Rest* (OvR) [2].

Dimensi kedua mengukur efisiensi komputasi fisik perangkat keras meliputi hitungan total kuantitas parameter aktif (dalam juta), kapasitas file penyimpanan biner akhir (dalam MB), serta durasi latensi kecepatan inferensi rata-rata sistem (dalam milidetik per citra, ms/img) [10] [14] [22].

3.6. Analisis Hasil

Tahap keenam diarahkan untuk membedah karakteristik keberhasilan komparatif maupun kesalahan biner model secara mendalam berdasarkan hasil pengujian. Analisis hasil dikerjakan secara kualitatif dan kuantitatif melalui visualisasi matriks konfusi (*confusion matrix*) untuk memetakan sensitivitas deteksi dan pola penyebaran prediksi salah (*misclassification*) antar kelas target. Hasil tersebut dilanjutkan ke dalam tahapan analisis kesalahan (*error analysis*) secara mendalam guna mengidentifikasi jenis citra kerusakan dinding yang gagal dikenali

model beserta tingkat kepercayaannya (*confidence score*) [2].

Pembedahan interpretasi keputusan arsitektur diperkuat lewat implementasi metode *Gradient-weighted Class Activation Mapping* (Grad-CAM) [2] [15] [18].

Metode ini mengekstraksi nilai gradien pada lapisan konvolusi terakhir untuk menghasilkan representasi visual peta panas (*heatmap*) intensitas piksel, sehingga memperlihatkan area koordinat spasial pada permukaan dinding yang paling memengaruhi keputusan prediksi dari masing-masing model CNN.

3.7. Rekomendasi Arsitektur

Prosedur akhir dari metodologi penelitian ini adalah merumuskan keputusan penentuan arsitektur model CNN yang paling optimal berdasarkan hasil komparasi parameter performa klasifikasi dan efisiensi fisik *hardware* [17].

Rekomendasi ini ditujukan sebagai panduan teknis implementasi sistem di lapangan pasca-bencana. Berdasarkan matriks keputusan akhir,

arsitektur MobileNetV3-Large direkomendasikan secara mutlak sebagai pilihan terbaik untuk kebutuhan implementasi pada perangkat komputer tepi terintegrasi (*mobile/edge deployment*) seperti aplikasi gawai pintar lapangan atau sistem mikro wahana tanpa awak (UAV) karena keunggulan efisiensi kompresi memori dan kecepatan latensinya yang ultra-ringan [6] [14] [20] [22].

Di sisi lain, arsitektur ResNet50 direkomendasikan khusus untuk skenario penggunaan yang memerlukan tingkat akurasi maksimum pada server pusat atau komputasi awan (*cloud server*) di mana ketersediaan pasokan daya perangkat keras tidak dibatasi [11] [15] [17] [21].

4. HASIL DAN PEMBAHASAN

4.1. Hasil Akhir Pelatihan dan Analisis Kurva Pembelajaran

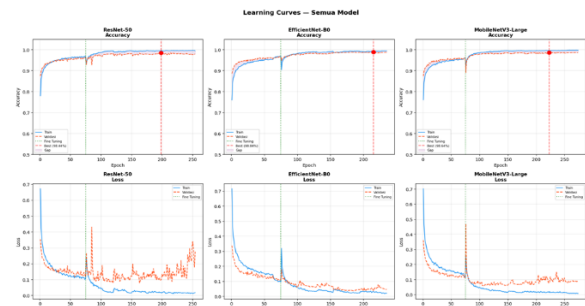
Proses pelatihan dievaluasi berdasarkan jumlah epoch yang dibutuhkan hingga mencapai konvergensi melalui mekanisme *early stopping*. Ringkasan hasil akhir pelatihan dari ketiga model disajikan pada tabel berikut.

Tabel 1. Hasil Akhir Pelatihan Ketiga Model

Model	Fase 1 (Epoch)	Best Val Acc (F1)	Fase 2 Aktual (Epoch)	Total Epoch F1 dan F2	Best Val Acc (F2)
ResNet50	75/75	96,18%	179/225	Epoch ke-254	98,44%
EfficientNet-B0	75/75	96,59%	160/225	Epoch ke-235	98,84%
MobileNetV3-Large	75/75	96,01%	199/225	Epoch ke-274	98,64%

Berdasarkan Tabel tersebut, EfficientNet-B0 mencapai konvergensi paling cepat, yaitu pada epoch ke-235 dengan akurasi validasi tertinggi sebesar 98,84%. Sebaliknya, MobileNetV3-Large memerlukan proses pelatihan terpanjang hingga 274 epoch sebelum mencapai titik jenuh. Perolehan akurasi validasi tertinggi EfficientNet-B0 (98,84%) berbanding terbalik dengan hasil pada test set independen, di mana model ini justru mencatatkan akurasi terendah (95,77%). Kesenjangan ini mengindikasikan *overfitting* yang lebih besar terhadap karakteristik data validasi, kemungkinan akibat mekanisme *compound scaling* pada EfficientNet-B0 yang mengoptimalkan kedalaman, lebar, dan resolusi berdasarkan koefisien hasil pencarian pada dataset ImageNet, sehingga arsitektur ini lebih sensitif terhadap distribusi augmentasi buatan pada data validasi dibandingkan variasi citra riil pada *test set*. Temuan ini menegaskan pentingnya evaluasi pada data uji independen, bukan hanya akurasi validasi, sebagai dasar pemilihan arsitektur terbaik.

Analisis terhadap kurva pembelajaran menunjukkan adanya lonjakan *validation loss* sementara pada epoch ke-75 yang merupakan dampak dari transisi dari fase 1 (*feature extraction*) ke fase 2 (*fine-tuning*) ketika seluruh bobot jaringan dibuka untuk dilatih kembali.



Gambar 2. Grafik perbandingan *loss* dan *accuracy*

Model ResNet50 menunjukkan fluktuasi *validation loss* yang lebih besar pada fase 2 karena jumlah parameternya yang besar membuat model lebih sensitif terhadap pembaruan bobot, sedangkan EfficientNet-B0 dan MobileNetV3-Large menunjukkan grafik yang jauh lebih stabil.

4.2. Evaluasi pada Data Pengujian (*Test set*)

Setelah pelatihan selesai, ketiga model diuji menggunakan *test set* yang berisi 3.000 citra independen yang belum pernah dilihat oleh model sebelumnya. Hasil evaluasi performa menyeluruh disajikan pada Tabel 2.

Tabel 2. Hasil Evaluasi Kinerja Ketiga Model

Metrik	ResNet50	EfficientNet-B0	MobileNetV3-Large
Accuracy	99,40%	95,77%	99,37%
Precision	99,40%	96,24%	99,37%
Recall	99,40%	95,77%	99,37%
F1-Score	99,40%	95,85%	99,37%
Cohen's Kappa	0,9928	0,9492	0,9924
Jumlah Salah	18 dari 3.000	127 dari 3.000	19 dari 3.000
Error Rate	0,60%	4,23%	0,63%

Tabel tersebut menunjukkan hasil yang berbeda dari fase validasi. ResNet50 menghasilkan akurasi pengujian tertinggi sebesar 99,40%, disusul sangat ketat oleh MobileNetV3-Large sebesar 99,37% (selisih 0,03%). Sementara itu, EfficientNet-B0 menghasilkan performa terendah dengan tingkat kesalahan mencapai 4,23% atau sebanyak 127 citra salah diklasifikasikan.

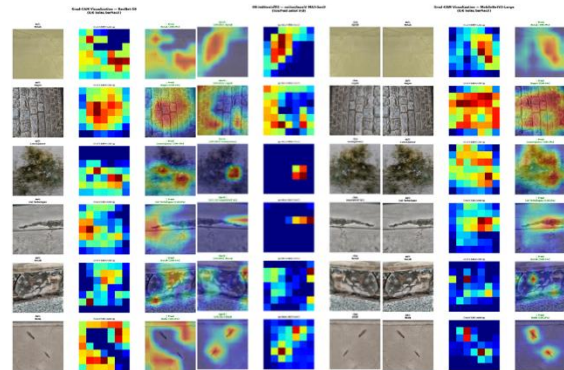
Berdasarkan analisis *classification report* per kelas, performa rendah EfficientNet-B0 disebabkan oleh kecenderungan *over-predict* secara sistematis pada kelas cat terkelupas, yang menurunkan nilai presisinya menjadi 0,83. Sebaliknya, ResNet50 dan MobileNetV3-Large menunjukkan distribusi nilai *F1-Score* yang sangat seimbang dan tinggi (0,99) di hampir seluruh kelas kerusakan.

4.3. Analisis Kesalahan dan Interpretasi Grad-CAM

Melalui analisis *confusion matrix*, ditemukan bahwa kesalahan klasifikasi dominan terjadi pada batas visual yang ambigu antara kelas retak dan cat terkelupas. Hal ini disebabkan oleh adanya pola linier halus yang menyerupai retakan pada permukaan cat yang terkelupas.

Visualisasi Grad-CAM digunakan untuk menganalisis area perhatian (*attention*) dari masing-masing model dalam mengekstraksi fitur kerusakan. Pertama model ResNet50 menampilkan area perhatian yang luas (*broad attention*), mencakup objek kerusakan utama sekaligus konteks spasial di

sekitarnya. Kedua EfficientNet-B0 menunjukkan fokus yang terlalu sempit dan terlokalisasi (*highly localized*), sehingga rentan gagal jika fitur kerusakan meluas keluar dari area fokus utama. Ketiga MobileNetV3-Large yang berhasil menunjukkan distribusi perhatian yang paling seimbang antara deteksi tekstur lokal dan konteks dinding secara umum.



Gambar 3. Perbandingan visualisasi heatmap Grad-CAM

Temuan ini konsisten dengan *confusion matrix*: dari 127 kesalahan EfficientNet-B0, mayoritas berasal dari *over-predict* ke kelas Cat Terkelupas (Lumut/Jamur 35 citra, Retak 28 citra, Rusak 28 citra), akibat area perhatian (*attention*) modelnya berdasarkan Grad-CAM yang terlalu terlokalisasi sehingga tekstur halus antar kelas sulit dibedakan. Sebaliknya, fokus ResNet50 yang lebih luas membatasi kesalahan hanya pada 18 citra (dominan Retak menjadi Cat Terkelupas, 4 citra), sementara keseimbangan fokus MobileNetV3-Large menekan kesalahan menjadi 19 citra tanpa pola sistematis, menunjukkan bahwa cakupan fokus, bukan jumlah parameter, menentukan akurasi klasifikasi.

4.4. Efisiensi Komputasi dan Rekomendasi Arsitektur

Untuk menentukan model yang paling siap diimplementasikan pada perangkat lapangan dengan sumber daya terbatas, efisiensi komputasi dari ketiga arsitektur dibandingkan pada Tabel 3.

Tabel 3. Perbandingan Kinerja Arsitektur Model Deep Learning

Arsitektur Model	Jumlah Parameter	Ukuran File Model (MB)	Kecepatan Inferensi (ms/citra)	Akurasi Pengujian
ResNet50	24,15 Juta	207,18 MB	336,14 ms	99,40%
EfficientNet-B0	4,42 Juta	31,67 MB	49,62 ms	95,77%
MobileNetV3-Large	3,28 Juta	27,34 MB	49,29 ms	99,37%

Meskipun ResNet50 memiliki akurasi tertinggi, model ini berukuran $7,57\times$ lebih besar (207,18 MB) dan $6,82\times$ lebih lambat (336,14 ms/citra) dibandingkan MobileNetV3-Large.

Berdasarkan analisis menyeluruh ini, MobileNetV3-Large merupakan arsitektur yang paling direkomendasikan untuk sistem inspeksi kerusakan

bangunan otomatis pasca-bencana berbasis perangkat *mobile*. Arsitektur ini mampu memotong kebutuhan memori secara drastis (hanya 27,34 MB) dan melakukan inferensi secara *real-time* (49,29 ms/citra atau ~ 20 frame per detik) dengan hanya mengorbankan akurasi sebesar 0,03% dari model ResNet50.

5. KESIMPULAN DAN SARAN

Penelitian ini berhasil membandingkan kinerja ResNet-50, EfficientNet-B0, dan MobileNetV3-Large untuk klasifikasi enam kategori kerusakan dinding pasca-gempa menggunakan 30.000 citra. ResNet-50 mencapai akurasi tertinggi (99,40%) dengan interpretasi Grad-CAM yang menunjukkan ekstraksi konteks spasial luas, namun berukuran besar (207,18 MB) dan paling lambat (336,14 ms/citra). MobileNetV3-Large direkomendasikan sebagai model paling optimal untuk lapangan karena hanya berukuran 27,34 MB dan berinferensi real-time (49,29 ms/citra) dengan penurunan akurasi tidak signifikan (0,03%; akurasi 99,37%). EfficientNet-B0 menunjukkan performa terendah (95,77%) akibat over-predict sistematis pada kelas cat terkelupas.

Meskipun MobileNetV3-Large menunjukkan hasil yang menjanjikan untuk aplikasi lapangan, penelitian ini tetap memiliki beberapa keterbatasan yang mendasari arah pengembangan selanjutnya. Pertama, mayoritas dataset merupakan augmentasi sintetis dari basis citra lapangan yang terbatas (5.987 citra) dan hanya dari satu wilayah (Sulawesi Utara), sehingga generalisasi lintas-wilayah belum teruji; disarankan perluasan dataset ke wilayah bencana lain. Kedua, pengujian efisiensi komputasi masih bersifat simulasi, belum diuji langsung pada perangkat mobile/UAV di lapangan; disarankan pengujian implementasi nyata pada perangkat tersebut. Ketiga, masih terdapat ambiguitas visual antara kelas retak dan cat terkelupas; disarankan penambahan fitur pembeda (misalnya tekstur) untuk mengurangnya. Keempat, ukuran model masih dapat dioptimalkan lebih lanjut untuk perangkat terbatas; disarankan penerapan kuantisasi bobot atau pruning untuk mempercepat inferensi.

DAFTAR PUSTAKA

- [1] Badan Meteorologi Klimatologi dan Geofisika, "Katalog Gempa Bumi dan Tsunami - Gempa Bumi," BMKG. Accessed: May 12, 2026. [Online]. Available: <https://www.bmkg.go.id/gempabumi/mitigasi/katalog-gempabumi-tsunami>
- [2] S. Wang, M. Li, and D. Le, "Structural damage detection and safety assessment method based on machine vision and machine learning," *PLoS One*, vol. 21, no. 2, p. e0341653, Feb. 2026, doi: 10.1371/journal.pone.0341653.
- [3] Ricky Putra Sardika and W. Widhiarso, "Klasifikasi Otomatis Tingkat Kerusakan Retak Bangunan pada Citra Digital Menggunakan MobileNetV2 dan Augmentasi Data," *Arcitech J. Comput. Sci. Artif. Intell.*, vol. 5, no. 1, pp. 108–124, Jun. 2025, doi: 10.29240/arcitech.v5i1.13938.
- [4] Badan Meteorologi Klimatologi dan Geofisika, "Ulasan Guncangan Tanah Akibat Gempa Bumi Maluku Utara 2 April 2026," BMKG. Accessed: May 02, 2026. [Online]. Available: <https://www.bmkg.go.id/gempabumi/ulasan-guncangan-tanah-akibat-gempabumi-maluku-utara-2-april-2026>
- [5] P. D. Ogunjinmi, S. Park, B. Kim, and D.-E. Lee, "Rapid Post-Earthquake Structural Damage Assessment Using Convolutional Neural Networks and Transfer Learning," *Sensors*, vol. 22, no. 9, p. 3471, May 2022, doi: 10.3390/s22093471.
- [6] N. Alsaaran and A. Soudani, "Deep Learning Image-Based Classification for Post-Earthquake Damage Level Prediction Using UAVs," *Sensors*, vol. 25, no. 17, p. 5406, Sep. 2025, doi: 10.3390/s25175406.
- [7] R. E. Philip, A. D. Andrushia, A. Nammalvar, B. G. A. Gurupatham, and K. Roy, "A Comparative Study on Crack Detection in Concrete Walls Using Transfer Learning Techniques," *J. Compos. Sci.*, vol. 7, no. 4, p. 169, Apr. 2023, doi: 10.3390/jcs7040169.
- [8] C. L. Nguyen, A. Nguyen, J. Brown, T. Byrne, B. T. Ngo, and C. X. Luong, "Optimising Concrete Crack Detection: A Study of Transfer Learning with Application on Nvidia Jetson Nano," *Sensors*, vol. 24, no. 23, p. 7818, Dec. 2024, doi: 10.3390/s24237818.
- [9] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jun. 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90.
- [10] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," *36th Int. Conf. Mach. Learn. ICML 2019*, vol. 2019-June, pp. 10691–10700, 2019, doi: 10.48550/arXiv.1905.11946.
- [11] A. Howard *et al.*, "Searching for MobileNetV3," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, IEEE, Oct. 2019, pp. 1314–1324. doi: 10.1109/ICCV.2019.00140.
- [12] T. Shahriar, "Comparative Analysis of Lightweight Deep Learning Models for Memory-Constrained Devices," Sep. 09, 2025, *arXiv*. doi: 10.48550/arXiv.2505.03303.
- [13] V. Singh *et al.*, "A Hybrid Deep Learning Model for Enhanced Structural Damage Detection: Integrating ResNet50, GoogLeNet, and Attention Mechanisms," *Sensors*, vol. 24, no. 22, p. 7249, Nov. 2024, doi: 10.3390/s24227249.
- [14] Z. Hong *et al.*, "Classification of Building Damage Using a Novel Convolutional Neural Network Based on Post-Disaster Aerial Images," *Sensors*, vol. 22, no. 15, p. 5920, Aug. 2022, doi: 10.3390/s22155920.
- [15] M. A. Ghofur and A. B. Ulum, "Analisis Perbandingan Kinerja Model ResNet50 Dan EfficientNet-B0 Dalam Klasifikasi Kerusakan Jalan," *Jumin*, vol. 6, no. 5, pp. 2504–2511, 2025, doi: 10.55338/jumin.v6i5.6843.
- [16] D. I. Mulyana and I. Wahyudi, "Deteksi

- Kerusakan Jalan Berdasarkan Citra Digital Menggunakan Convolutional Neural Network (CNN)," *J. Indones. Manaj. Inform. dan Komun.*, vol. 6, no. 1, pp. 294–302, Jan. 2025, doi: 10.35870/jimik.v6i1.1192.
- [17] A. N. Abiyyu and M. Rahardi, "Comparison of Transfer learning Models MobileNetV3-Large and EfficientNet-B0 for Rice Leaf Disease Classification," *J. Appl. Informatics Comput.*, vol. 10, no. 1, pp. 818–828, Feb. 2026, doi: 10.30871/jaic.v10i1.12033.
- [18] A. R. Dianto, F. T. A. -, and H. M. -, "ANALISIS EFEKTIFITAS ALGORITMA MOBILENETV3-LARGE DAN EFFICIENTNET-B0 UNTUK KLASIFIKASI CITRA PENYAKIT DAUN JERUK," *J. Inform. dan Tek. Elektro Terap.*, vol. 13, no. 3, pp. 697–705, Jul. 2025, doi: 10.23960/jitet.v13i3.6956.
- [19] K. Kansal, T. B. Chandra, and A. Singh, "ResNet-50 vs. EfficientNet-B0: Multi-Centric Classification of Various Lung Abnormalities Using Deep Learning," *Procedia Comput. Sci.*, vol. 235, pp. 70–80, 2024, doi: 10.1016/j.procs.2024.04.007.
- [20] L. Chen *et al.*, "Evaluation of MobileNetV3-Large for crack classification across low- and high-resolution images," *Front. Built Environ.*, vol. 11, no. December, Dec. 2025, doi: 10.3389/fbuil.2025.1724879.
- [21] M. N. Razali, N. Arbaiy, P. Lin, and S. Ismail, "Optimizing Multiclass Classification Using Convolutional Neural Networks with Class Weights and Early Stopping for Imbalanced Datasets," *Electronics*, vol. 14, no. 4, p. 705, Feb. 2025, doi: 10.3390/electronics14040705.
- [22] A. Raza, F. Hanif, and H. A. Mohammed, "Efficient crack and surface-type recognition via CNN-block development mechanism and edge profiling," *Sci. Rep.*, vol. 15, no. 1, p. 40073, Nov. 2025, doi: 10.1038/s41598-025-25956-8.
- [23] G. M. van de Ven, N. Soures, and D. Kudithipudi, "Continual learning and catastrophic forgetting," in *Learning and Memory: A Comprehensive Reference*, 2nd Editio., Elsevier, 2025, pp. 153–168. doi: 10.1016/B978-0-443-15754-7.00073-0.