

DETEKSI OBJEK BERBAHAYA PADA CITRA X-RAY DI BANDARA MENGUNAKAN MODEL *RETINANET* BERBASIS *RESNET50*

Damar Riyadi Syahputra^{1*}, Dedy Kurniadi²

^{1,2}Universitas Islam Sultan Agung, Indonesia

32602200052@std.unissula.ac.id^{1*}

ddy.kurniadi@unissula.ac.id²

*corresponding author

Received: 20-04- 2026

Revised: 15-05-2026

Approved: 26-05-2026

ABSTRAK

Keamanan bandara sangat bergantung pada pemeriksaan bagasi menggunakan pemindai X-ray. Namun, faktor kelelahan petugas dan tumpang tindih barang (*occlusion*) memunculkan risiko *human error*. Penelitian ini bertujuan mengimplementasikan model deteksi objek otomatis menggunakan arsitektur *RetinaNet* dengan tulang punggung (*backbone*) *Resnet50* berbasis *Convolutional Neural Network (CNN)* untuk mengidentifikasi objek berbahaya berupa Senjata Api, Pisau, Tang, Gunting, dan Kunci Inggris pada citra X-ray koper. Dataset yang digunakan adalah kumpulan citra publik *SIXray* sebanyak 8.295 citra. Pelatihan model menerapkan strategi *two-stage training* (*pembekuan backbone* dan *fine tuning*) serta memanfaatkan *Focal Loss* untuk mengatasi ketidakseimbangan kelas. Hasil prediksi kemudian disaring menggunakan algoritma *Non-Maximum Suppression (NMS)*. Hasil pengujian menunjukkan model mencapai nilai *Mean Average Precision (map)* sebesar 43,71% pada *IoU* 50%, dengan *map* keseluruhan 23,37% dan *Recall* keseluruhan 32,07%. Model mendeteksi objek besar secara optimal dengan *Recall* 38,27%, namun masih kesulitan pada objek berukuran kecil akibat oklusi ekstrem dengan nilai *Recall* 1,43%. Secara kualitatif, model berhasil mengenali pola ancaman, meski masih memunculkan tumpang tindih *bounding box* dan beberapa *false positive* pada latar belakang. Kesimpulannya, model *Resnet50-RetinaNet* memiliki potensi sebagai sistem pendamping otomatis di bandara, namun memerlukan penambahan sampel latar belakang dan penyesuaian parameter *NMS* lebih ketat untuk meningkatkan presisi.

Kata kunci: Citra X-Ray, Deteksi Objek, *Focal Loss*, *RetinaNet*, *ResNet50*

PENDAHULUAN

Keamanan bandara merupakan aspek penting dalam sistem transportasi global karena berkaitan langsung dengan keselamatan penumpang, awak pesawat, serta infrastruktur penerbangan. Bandara juga rentan terhadap berbagai ancaman, seperti penyelundupan barang terlarang, Senjata Api, Pisau, Tang, Gunting, dan Kunci Inggris (Marsya Aulia & Dyahjatmayanti, 2024). Oleh karena itu, sistem keamanan bandara dituntut mampu melakukan deteksi ancaman secara cepat dan akurat guna meminimalkan risiko yang dapat membahayakan keselamatan penerbangan maupun keamanan nasional (Kusumadewi et al., 2023).

Seiring perkembangan teknologi, penggunaan perangkat X-Ray telah menjadi bagian penting dalam proses pemeriksaan barang bawaan penumpang di bandara. Teknologi ini memungkinkan petugas keamanan melakukan pemeriksaan isi bagasi tanpa perlu membukanya secara fisik sehingga proses inspeksi menjadi lebih cepat dan efisien (Abu Ayub, 2021). Selain itu, perkembangan teknologi X-Ray juga mampu menghasilkan citra dengan kualitas resolusi dan kontras yang lebih baik sehingga objek mencurigakan dapat diidentifikasi secara lebih jelas. Meskipun demikian, proses interpretasi citra hasil pemindaian masih sangat bergantung pada kemampuan manusia dalam

menganalisis objek yang ditampilkan pada layar pemindaian.

Ketergantungan terhadap kemampuan manusia dalam proses interpretasi citra X-Ray menimbulkan berbagai tantangan, seperti kelelahan kerja, keterbatasan waktu pemeriksaan, serta faktor subjektivitas dalam menganalisis citra. Kondisi tersebut berpotensi menyebabkan *human error* yang dapat menurunkan tingkat akurasi deteksi ancaman dan berdampak pada keselamatan penerbangan (Sumiyati & Kifni Yudianto, 2024) . Permasalahan ini menjadi tantangan utama dalam sistem keamanan bandara modern dan menunjukkan perlunya dukungan teknologi yang mampu membantu proses interpretasi secara otomatis, cepat, dan konsisten.

Penelitian mengenai deteksi objek berbahaya pada citra X-Ray telah banyak dilakukan menggunakan berbagai pendekatan *Deep Learning* dan berbagai metode telah dikembangkan untuk meningkatkan performa deteksi, seperti penggunaan *Mask R-CNN* dengan *Softer-NMS* yang mampu meningkatkan nilai mAP pada citra X-Ray kompleks (Rahbari et al., 2017) . Selain itu, pendekatan berbasis *YOLO* dan jaringan berbasis attention mechanism juga menunjukkan peningkatan performa deteksi pada objek berbahaya di citra X-Ray (Yang et al., 2025) . Meskipun demikian, sebagian besar penelitian terdahulu masih menghadapi kendala dalam mendeteksi objek berukuran kecil dan objek yang mengalami tumpang tindih tinggi pada bagasi.

Penelitian lain juga menunjukkan bahwa performa deteksi pada dataset SIXray sangat dipengaruhi oleh kondisi *occlusion* dan distribusi data yang tidak seimbang. Penggunaan arsitektur berbasis *Feature Pyramid Network* (FPN), attention mechanism, hingga *Transformer* dikembangkan untuk meningkatkan kemampuan deteksi multi-skala dan memperbaiki representasi fitur objek kecil (Viriyasaranon et al., 2022). Namun, kompleksitas arsitektur yang semakin tinggi menyebabkan kebutuhan komputasi dan proses pelatihan menjadi lebih besar. Oleh karena itu, penelitian ini mengusulkan penggunaan *RetinaNet* dengan *backbone ResNet50* sebagai pendekatan yang lebih ringan namun tetap mampu menghasilkan performa deteksi yang baik pada citra X-Ray (Rahmawati et al., 2025) . Selain itu, penggunaan *Focal Loss* pada *RetinaNet* diharapkan dapat membantu mengatasi permasalahan class imbalance yang menjadi tantangan utama pada *dataset SIXray*.

METODE PENELITIAN

Preprocessing Data

Tahapan preprocessing data merupakan langkah awal yang sangat penting dalam proses pembangunan model deteksi, karena kualitas data akan berpengaruh langsung terhadap kinerja model (Aprianto et al., 2025) . Pada penelitian ini, preprocessing dilakukan melalui lima tahapan utama, yakni :

1. Pemetaan label (*Label mapping*), mengonversi label kelas dari bentuk teks menjadi nilai integer untuk dapat menghitung *loss function* dan mengoptimalkan prediksi klasifikasi.
2. Pengubahan ukuran (*Resizing*), gambar diubah ukurannya menjadi dimensi 416×416 piksel untuk meminimalkan beban komputasi.
3. Penyesuaian skala *bounding box* (*Bounding box rescaling*), hal ini dilakukan untuk penyesuaian skala koordinat *bounding box* setelah dilakukan tahap

resizing yang tentu hal tersebut mengubah dimensi citra. Penyesuaian dilakukan dengan rumus berikut:

Untuk sumbu X (koordinat horizontal: x_{min}, x_{max}):

$$x_{baru} = x_{lama} \times \frac{416}{w} \quad (1)$$

Untuk sumbu Y (koordinat vertikal: y_{min}, y_{max}):

$$y_{baru} = y_{lama} \times \frac{416}{h} \quad (2)$$

Di mana:

x_{baru} dan y_{baru} = koordinat *bounding box* setelah citra di-*resize*.

x_{lama} dan y_{lama} = koordinat *bounding box* asli sebelum citra di-*resize*.

416 = dimensi target citra yang baru (piksel).

w dan h = lebar (*width*) dan tinggi (*height*) citra asli.

4. *Augmentasi (Augmentation)*, dilakukan untuk memperkaya variasi citra pada *training set* (Sanjaya & Ayub, 2020). Transformasi yang diterapkan menggunakan beberapa teknik modifikasi spasial secara acak dengan probabilitas tertentu, yang meliputi:
 - a. Pembalikan citra secara horizontal (*Horizontal Flip*) dengan probabilitas 50%.
 - b. Pembalikan citra secara vertikal (*Vertical Flip*) dengan probabilitas 50%.
 - c. Rotasi acak kelipatan 90 derajat (*Random Rotate 90*) dengan probabilitas 50%.
 - d. Transformasi *Affine* dengan probabilitas 80%, yang mencakup perbesaran skala citra (*scaling*) antara 1,0 hingga 1,3 kali lipat dan rotasi acak pada rentang -15 hingga 15 derajat, dengan tetap mempertahankan rasio aspek (*aspect ratio*) citra.
5. Konversi ke Format TFRecord, mengubah citra beserta anotasinya (*filename*, koordinat *bounding box*, dan kelas objek) untuk diserialisasi menjadi format file biner TFRecord. Konversi ini bertujuan untuk mengoptimalkan *pipeline input* data pada ekosistem *TensorFlow*, sehingga proses pembacaan data dalam jumlah besar selama pelatihan model dapat berjalan jauh lebih cepat dan efisien tanpa terhambat oleh proses *I/O* (Input/Output).

Convolutional Neural Network

Convolutional Neural Network (CNN) merupakan salah satu arsitektur *Deep Learning* yang dirancang untuk memproses data berbentuk grid dua dimensi, seperti citra digital (Hartato, 2021). Algoritma CNN dapat digunakan untuk membedakan gambar satu dengan gambar yang lain (Firmansyah et al., 2025; Harahap et al., 2022). Hal ini karena CNN memiliki keunggulan dibandingkan jaringan saraf tiruan tradisional karena menggunakan konsep konektivitas lokal (*local connectivity*) dan pembagian parameter (*parameter sharing*), sehingga proses komputasi menjadi lebih efisien dan mampu mempertahankan informasi spasial pada citra. Melalui proses pembelajaran bertingkat, CNN dapat mengekstraksi fitur visual secara hierarkis, mulai dari fitur sederhana seperti garis dan tepi hingga fitur kompleks berupa bentuk objek secara utuh (Jinan & Hayadi, 2022).

Pada penelitian ini, arsitektur CNN digunakan sebagai dasar dalam proses

ekstraksi fitur citra X-Ray untuk mendukung proses identifikasi objek. Secara umum, CNN terdiri atas dua bagian utama, yaitu *feature extraction* dan *classification*. Tahap *feature extraction* meliputi *Convolutional Layer*, fungsi aktivasi, dan *Pooling Layer*, sedangkan tahap *classification* digunakan untuk menentukan kelas objek berdasarkan fitur yang telah diperoleh.

Convolutional Layer merupakan komponen utama pada CNN yang berfungsi melakukan ekstraksi fitur dari citra masukan. Proses ini dilakukan dengan menerapkan filter atau kernel yang digeser pada seluruh area citra untuk menghasilkan *feature map*. Operasi konvolusi dapat dirumuskan sebagai berikut:

$$S(i, j) = (I * K)(i, j) = \sum_m \sum_n I(i - m, j - n)K(m, n) \quad (3)$$

Pada persamaan tersebut, $(S(i, j))$ merupakan nilai piksel pada *feature map* di koordinat tertentu, (I) adalah citra masukan, sedangkan (K) merupakan kernel atau filter yang digunakan. Variabel (m) dan (n) menunjukkan dimensi kernel yang diterapkan pada citra.

Hasil dari proses konvolusi kemudian diteruskan ke fungsi aktivasi untuk menambahkan sifat non-linear pada jaringan. Fungsi aktivasi yang umum digunakan pada CNN adalah *Rectified Linear Unit* (ReLU) karena mampu mengurangi permasalahan vanishing gradient pada jaringan yang dalam. Fungsi ReLU dirumuskan sebagai berikut:

$$f(x) = \max(0, x) \quad (4)$$

Pada persamaan tersebut, $(f(x))$ merupakan nilai keluaran setelah proses aktivasi, sedangkan (x) adalah nilai hasil konvolusi. Fungsi ReLU akan mempertahankan nilai positif dan mengubah nilai negatif menjadi nol. Proses ini membantu jaringan dalam mempelajari pola kompleks pada data citra (Hastomo & Satyo Bayangkari Karno, 2021; Herdianto & Nasution, 2022).

Setelah melewati fungsi aktivasi, *feature map* diproses pada *Pooling Layer* untuk melakukan reduksi dimensi spasial (*downsampling*). Tahapan ini bertujuan mengurangi kompleksitas komputasi dan meminimalkan risiko *overfitting*. Metode pooling yang umum digunakan adalah *Max Pooling*, yaitu dengan mengambil nilai terbesar dari suatu wilayah fitur. Operasi *Max Pooling* dapat dirumuskan sebagai berikut:

$$y = \max_{i, j \in R} x_{i, j} \quad (5)$$

Pada persamaan tersebut, (y) merupakan hasil pooling, sedangkan $(x_{\{(i, j)\}})$ adalah nilai piksel pada wilayah spasial (R) . Proses ini membantu model mempertahankan fitur penting sekaligus meningkatkan ketahanan terhadap pergeseran atau perubahan posisi objek pada citra.

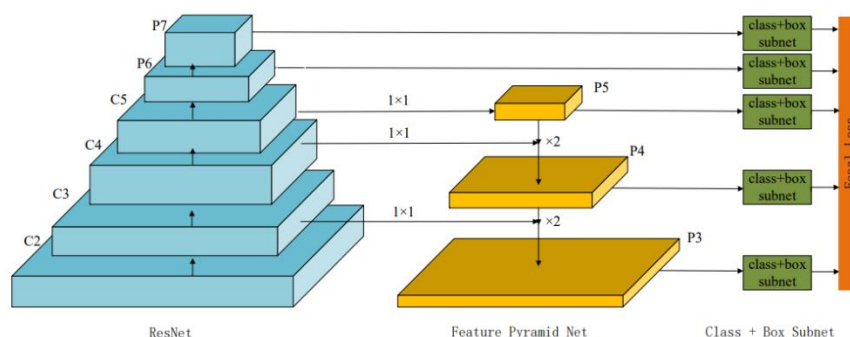
Pada arsitektur pengenalan citra konvensional, keluaran dari proses ekstraksi fitur selanjutnya diteruskan menuju *Fully Connected Layer* untuk proses klasifikasi. Namun, pada sistem deteksi objek modern, bagian ekstraksi fitur CNN sering dimanfaatkan sebagai *backbone feature extractor*. *Feature map* yang dihasilkan backbone kemudian digunakan oleh arsitektur deteksi objek untuk menentukan lokasi dan kelas objek secara simultan. Pendekatan ini dinilai efektif dalam mendeteksi objek pada citra kompleks, termasuk citra hasil pemindaian X-Ray bandara (Nasution et al., 2025).

RetinaNet

RetinaNet merupakan arsitektur deteksi objek satu tahap (*one-stage detector*) yang dirancang untuk menghasilkan akurasi tinggi dengan tetap

mempertahankan kecepatan inferensi. Berbeda dengan metode deteksi dua tahap yang memerlukan proses region proposal secara terpisah, *RetinaNet* melakukan proses lokalisasi dan klasifikasi objek secara langsung dalam satu tahap komputasi. Pendekatan ini membuat *RetinaNet* lebih efisien dan sesuai digunakan pada sistem yang membutuhkan proses deteksi secara cepat, termasuk pada analisis citra X-Ray.

Pada penelitian ini, *RetinaNet* digunakan sebagai model deteksi objek untuk mengidentifikasi barang berbahaya pada citra X-Ray. Secara umum, arsitektur *RetinaNet* terdiri atas tiga komponen utama, yaitu *backbone network*, *Feature Pyramid Network* (FPN), dan dua sub-jaringan fungsional. *Backbone network* pada penelitian ini menggunakan *Resnet50* yang berfungsi mengekstraksi fitur visual dari citra masukan. *Feature map* yang dihasilkan kemudian diteruskan ke *Feature Pyramid Network* (FPN) untuk membangun representasi fitur pada berbagai skala.



Gambar 1. Arsitektur *RetinaNet*

Gambar tersebut menunjukkan arsitektur *RetinaNet* yang terdiri atas tiga komponen utama, yaitu backbone ResNet, Feature Pyramid Network (FPN), serta Class dan Box Subnet. Pada bagian backbone, ResNet berfungsi sebagai pengekstraksi fitur utama dari citra masukan melalui beberapa tingkatan feature map yang ditandai dengan C2, C3, C4, dan C5. Semakin dalam lapisan jaringan, informasi semantik yang dihasilkan menjadi semakin kaya, meskipun resolusi spasial citra mengalami penurunan. Feature map hasil ekstraksi dari backbone kemudian diteruskan menuju Feature Pyramid Network (FPN) untuk membangun representasi fitur multi-skala (Tian et al., 2020).

FPN bekerja dengan menggabungkan informasi dari beberapa tingkat *feature map* untuk menghasilkan lapisan piramida fitur P3, P4, P5, P6, dan P7. Proses ini dilakukan melalui mekanisme lateral connection menggunakan konvolusi 1×1 serta proses upsampling pada *feature map* tingkat atas. Dengan pendekatan tersebut, model mampu mempertahankan informasi semantik yang kuat sekaligus menjaga detail spasial objek, sehingga objek berukuran kecil maupun besar dapat dideteksi dengan lebih optimal.

Setelah proses pembentukan feature pyramid selesai, setiap feature map akan diproses oleh dua subnet yang bekerja secara paralel, yaitu *Classification Subnet* dan *Box Regression Subnet*. *Classification Subnet* berfungsi untuk memprediksi kelas objek pada setiap anchor box, sedangkan *Box Regression Subnet* digunakan untuk menentukan koordinat *bounding box* objek secara lebih presisi. Seluruh hasil prediksi kemudian dievaluasi menggunakan *Focal Loss* untuk mengatasi ketidakseimbangan data antara objek dan background selama proses pelatihan model. Kombinasi antara backbone *ResNet50*, FPN, dan subnet

deteksi tersebut menjadikan *RetinaNet* mampu menghasilkan performa deteksi objek yang cepat dan akurat pada berbagai ukuran objek.

Sementara itu, pada *Box Regression Subnet*, digunakan *Smooth L1 Loss* untuk menghitung tingkat kesalahan lokalisasi koordinat. Fungsi ini dipilih karena sifatnya yang lebih tangguh (*robust*) terhadap data pencilan (*outlier*). Persamaannya adalah sebagai berikut:

$$\text{Smooth}_{L1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherwise} \end{cases} \quad (6)$$

Di mana x merupakan selisih atau galat antara koordinat hasil prediksi model dengan koordinat asli (*ground truth*).

Resnet50

Resnet50 merupakan salah satu arsitektur *Convolutional Neural Network* (CNN) yang termasuk dalam keluarga *Residual Network* (*ResNet*). Arsitektur ini dikembangkan untuk mengatasi permasalahan degradasi akurasi pada jaringan saraf yang memiliki lapisan sangat dalam. Pada jaringan CNN konvensional, penambahan jumlah lapisan sering menyebabkan penurunan performa akibat terjadinya *vanishing gradient*, yaitu kondisi ketika nilai gradien semakin kecil selama proses *backpropagation* sehingga pembaruan bobot pada lapisan awal tidak berlangsung secara optimal. Untuk mengatasi permasalahan tersebut, *ResNet* memperkenalkan konsep *residual learning* melalui mekanisme *shortcut connection* atau *identity shortcut* (Awaludin et al., 2022).

Berdasarkan mekanisme *shortcut connection*, aliran informasi pada jaringan dapat melewati satu atau beberapa lapisan secara langsung sehingga informasi penting dari lapisan sebelumnya tetap terjaga. Pendekatan ini memungkinkan proses propagasi gradien berjalan lebih stabil dan membantu jaringan yang sangat dalam tetap dapat dilatih tanpa mengalami penurunan performa. Dengan demikian, *Resnet50* mampu mempertahankan akurasi model meskipun memiliki jumlah lapisan yang lebih banyak dibandingkan arsitektur CNN konvensional.

Secara arsitektur, *Resnet50* terdiri atas 50 lapisan yang menggunakan desain *bottleneck*. Setiap *bottleneck block* tersusun atas tiga lapisan konvolusi, yaitu konvolusi 1×1 , 3×3 , dan kembali ke 1×1 . Lapisan konvolusi 1×1 digunakan untuk mereduksi dan mengembalikan dimensi fitur, sedangkan lapisan 3×3 berfungsi melakukan ekstraksi fitur utama. Penggunaan desain *bottleneck* memungkinkan *Resnet50* melakukan ekstraksi fitur kompleks dengan efisiensi komputasi yang lebih baik. Selain itu, setiap lapisan konvolusi pada *Resnet50* dilengkapi dengan *Batch Normalization* untuk menjaga stabilitas distribusi data dan fungsi aktivasi ReLU untuk mempertahankan sifat non-linear jaringan.

Pada penelitian ini, *Resnet50* digunakan sebagai backbone *feature extractor* dalam proses pengolahan citra X-Ray. Alur pemrosesan data dimulai dari tahap input, yaitu citra masukan yang terlebih dahulu melewati proses *Zero Padding* untuk mempertahankan dimensi citra. Selanjutnya, citra diproses pada Stage 1 melalui lapisan *Convolution*, *Batch Normalization*, ReLU, dan *Max Pooling* untuk mengekstraksi fitur dasar. Proses ekstraksi fitur kemudian dilanjutkan pada Stage 2 hingga Stage 5 menggunakan kombinasi *Conv Block* dan *Identity Block* yang memanfaatkan *shortcut connection* agar informasi penting tetap dipertahankan pada jaringan yang dalam. Pada tahap akhir, *feature map* yang

dihasilkan diproses menggunakan *Average Pooling*, kemudian diratakan melalui proses *Flattening* sebelum diteruskan ke *Fully Connected Layer* untuk menghasilkan keluaran klasifikasi.

Dalam implementasinya, *Resnet50* memiliki kemampuan yang baik dalam mempelajari representasi fitur visual secara hierarkis, mulai dari pola sederhana hingga fitur objek yang kompleks. Oleh karena itu, *Resnet50* banyak digunakan sebagai *backbone* pada berbagai sistem deteksi objek berbasis transfer learning. Dengan memanfaatkan *pre-trained weights* dari dataset ImageNet, model dapat mengadaptasi pengetahuan fitur visual umum untuk mendeteksi objek spesifik pada domain lain, termasuk objek berbahaya pada citra X-Ray. Stabilitas performa dan efisiensi pelatihannya menjadikan *Resnet50* sebagai salah satu arsitektur yang banyak digunakan dalam pengembangan sistem deteksi berbasis kecerdasan buatan saat ini.

Feature Pyramid Network (FPN)

Feature Pyramid Network (FPN) merupakan arsitektur ekstraksi fitur yang digunakan untuk meningkatkan kemampuan deteksi objek pada berbagai skala ukuran citra. Dalam proses deteksi objek, salah satu tantangan utama adalah mengenali objek berukuran kecil tanpa mengurangi kemampuan model dalam mendeteksi objek berukuran besar. FPN dirancang untuk mengatasi permasalahan tersebut dengan membangun representasi fitur multi-skala yang memiliki informasi semantik kuat pada setiap tingkatannya. Arsitektur ini memanfaatkan struktur hierarkis dari jaringan *Convolutional Neural Network* untuk menghasilkan *feature map* yang saling terhubung dan terintegrasi (Zhu et al., 2022).

Pada penelitian ini, FPN digunakan sebagai bagian dari arsitektur *RetinaNet* untuk memperkuat hasil ekstraksi fitur dari *backbone Resnet50*. Secara umum, FPN bekerja melalui tiga mekanisme utama, yaitu *bottom-up pathway*, *top-down pathway*, dan *lateral connections*.

Bottom-up pathway merupakan jalur ekstraksi fitur utama yang berasal dari *backbone Resnet50*. Pada tahap ini, citra masukan diproses secara bertahap melalui beberapa lapisan konvolusi sehingga resolusi spasial *feature map* menjadi semakin kecil, namun informasi semantik yang diperoleh menjadi lebih kaya. *Feature map* pada lapisan terdalam memiliki kemampuan representasi objek yang baik, tetapi detail spasial objek kecil sering kali berkurang akibat penurunan resolusi.

Untuk mengatasi hilangnya detail spasial tersebut, FPN menggunakan *top-down pathway* yang berfungsi meningkatkan resolusi *feature map* melalui proses *upsampling*. *Feature map* dari lapisan yang lebih dalam diperbesar kembali agar memiliki ukuran spasial yang sesuai dengan *feature map* pada lapisan sebelumnya. Proses ini membantu mempertahankan informasi semantik sekaligus meningkatkan representasi detail objek.

Selain itu, FPN juga menerapkan *lateral connections* untuk menggabungkan *feature map* dari jalur *bottom-up* dengan *feature map* hasil *upsampling* dari jalur *top-down*. Penggabungan dilakukan melalui operasi penjumlahan elemen demi elemen sehingga informasi spasial dan semantik dapat dipadukan secara optimal. Dengan mekanisme tersebut, FPN mampu menghasilkan *feature map* multi-skala yang lebih efektif untuk mendeteksi objek dengan ukuran yang bervariasi pada citra X-Ray (Raghu et al., 2021).

Class Imbalance

Masalah ketidakseimbangan kelas (*class imbalance*) adalah hambatan utama dalam pelatihan *Deep Learning* (Syafarina et al., 2025) . Metode konvensional seperti oversampling atau undersampling sering kali tidak efektif pada kasus multi-label karena mengubah distribusi satu label dapat merusak distribusi label lainnya yang muncul bersamaan (*co-occurrence*) (Liu et al., 2021) . Untuk mengatasi permasalahan tersebut, digunakan fungsi loss khusus yaitu Focal Loss. Focal Loss dirancang untuk mengurangi kontribusi sampel yang mudah diklasifikasikan dan memberikan fokus lebih besar pada sampel sulit atau kelas minoritas. Dengan pendekatan ini, model dapat mempelajari objek penting secara lebih efektif sehingga meningkatkan stabilitas dan akurasi deteksi objek. Rumus *Focal Loss* didefinisikan sebagai:

$$FL(p_t) = -\alpha_t(1 - p_t)^\gamma \log(p_t) \quad (7)$$

Di mana p_t adalah probabilitas prediksi model, α_t adalah parameter pembobot untuk menyeimbangkan kelas, dan γ adalah parameter pemfokusan untuk mengurangi bobot pada sampel yang mudah ditebak.

Non-Maximum Suppression (NMS)

Non-Maximum Suppression (NMS) merupakan metode pasca-pemrosesan (*post-processing*) yang digunakan pada sistem deteksi objek untuk menyaring hasil prediksi *bounding box*. Pada proses inferensi, model deteksi objek seperti RetinaNet umumnya menghasilkan beberapa *bounding box* yang saling tumpang tindih pada objek yang sama dengan nilai *confidence score* yang berbeda. Kondisi tersebut dapat menyebabkan munculnya deteksi ganda sehingga hasil deteksi menjadi kurang efektif. Oleh karena itu, NMS digunakan untuk memilih *bounding box* terbaik berdasarkan nilai *confidence score* tertinggi dan menghapus *bounding box* lain yang memiliki tingkat tumpang tindih tinggi atau nilai *Intersection over Union* (IoU) melebihi ambang batas tertentu.

Pada penelitian ini, NMS diterapkan setelah proses prediksi objek untuk memastikan setiap objek pada citra X-Ray hanya direpresentasikan oleh satu *bounding box* yang paling akurat. Proses ini membantu mengurangi deteksi redundan pada objek yang sama sehingga hasil deteksi menjadi lebih bersih dan mudah diinterpretasikan. Dengan penerapan NMS, model mampu mempertahankan prediksi dengan probabilitas tertinggi dan lokasi *bounding box* yang paling tepat, sehingga meningkatkan presisi sistem deteksi objek pada citra X-Ray koper.

Evaluasi

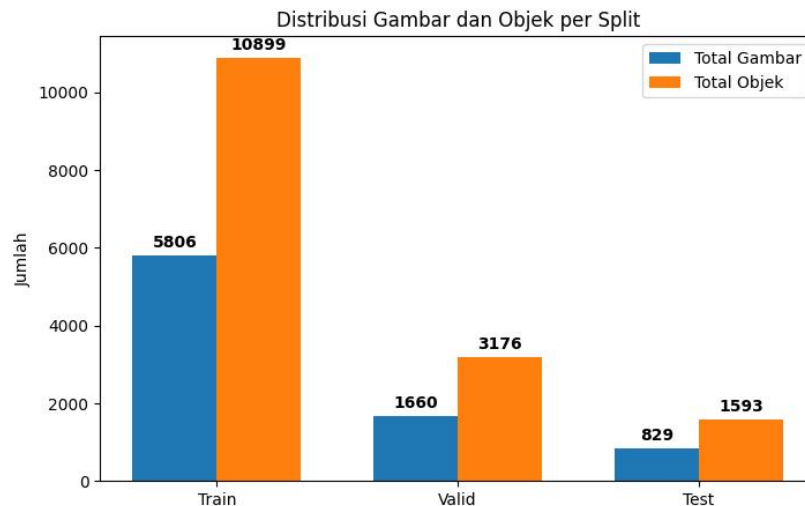
$$Precision = \frac{TP}{TP + FP} \quad (8)$$

$$Recall = \frac{TP}{TP + FN} \quad (9)$$

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (10)$$

HASIL DAN PEMBAHASAN

Penelitian ini menggunakan dataset sekunder yang diperoleh dari platform repositori Roboflow. Data tersebut mencakup kumpulan citra X-ray bagasi atau koper yang memuat objek berbahaya di dalamnya. Secara keseluruhan, dataset ini terdiri dari 8.295 citra X-ray dengan total 15.668 anotasi objek berbahaya.



Gambar 2. Distribusi Gambar dan Objek

Berdasarkan grafik distribusi pada gambar 2 dataset dibagi dengan rasio 70:20:10 dengan rincian sebagai berikut:

- *Training Set* (70%): 5806 gambar dan 10.899 objek berbahaya.
- *Validation Set* (20%): 1.660 gambar dan 3.176 objek berbahaya.
- *Test Set* (10%): 829 gambar dan 1593 objek berbahaya.

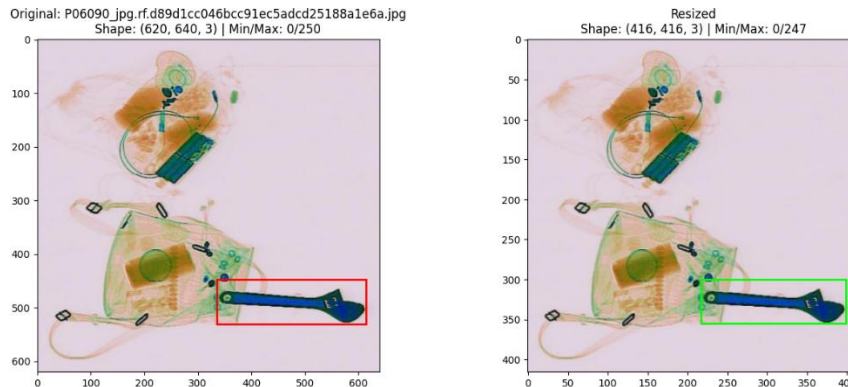
Preprocessing Data

1. Pemetaan label (*Label mapping*)

Tabel 1. Pemetaan Label Kelas

Kelas Teks (<i>String</i>)	Representasi Numerik (<i>Integer</i>)
<i>Gun</i>	0
<i>Knife</i>	1
<i>Pliers</i>	2
<i>Scissor</i>	3
<i>Wrench</i>	4

2. Penyesuaian ukuran dan koordinat (*Resize and rescaling*)



Gambar 3. Visualisasi Hasil *Resize* dan Penyesuaian *Bounding box*

Berdasarkan gambar 4, citra asli pada sisi kiri memiliki dimensi awal 620x640 dengan *bounding box* berwarna merah yang menandai objek Kunci Inggris (*Wrench*). Setelah proses *resize* menjadi 416x416 pada sisi kanan, objek berhasil dipertahankan proporsinya di dalam kanvas, dan *bounding box* baru (berwarna hijau) tetap membingkai objek secara presisi.

Tabel 2. Perbandingan Koordinat *Bounding box*

Original	620 x 640	[335, 448, 614, 530]
Resized	416 x 416	[217, 300, 399, 355]

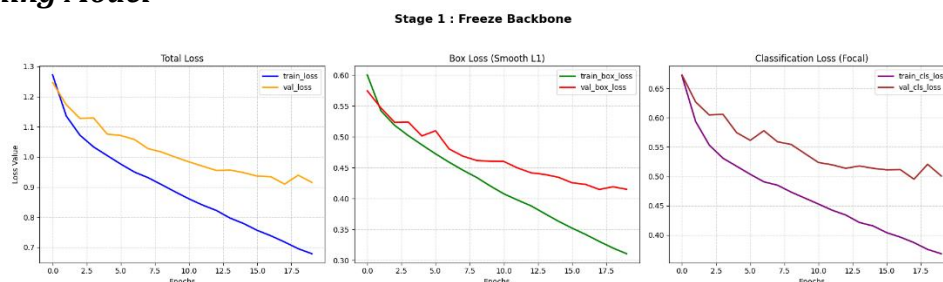
3. Augmentasi

Tabel 3. Parameter Augmentasi Data

Jenis Augmentasi	Parameter	Probabilitas
<i>Horizontal Flip</i>	-	50%
<i>Vertical Flip</i>	-	50%
<i>Random Rotate</i>	90°	50%
<i>Affine Transform</i>	<i>Scale</i> 1.0 s.d 1.3 <i>Rotate</i> -15° s.d 15° <i>Keep Ratio</i> : True	80%

Berdasarkan tabel 3, proses augmentasi menerapkan berbagai transformasi spasial secara acak berdasarkan nilai probabilitas, probabilitas sebesar 50% diberikan untuk membalikkan citra (*horizontal/vertical flip*) dan memutar citra secara tegak lurus (*rotate 90°*). Selain itu, transformasi *Affine* diaplikasikan dengan probabilitas sangat tinggi (80%) untuk mensimulasikan variasi ukuran objek (skala 1.0 hingga 1.3) dan kemiringan objek yang tidak beraturan di dalam tumpukan barang koper (rotasi kecil antara -15 hingga 15 derajat).

Training Model

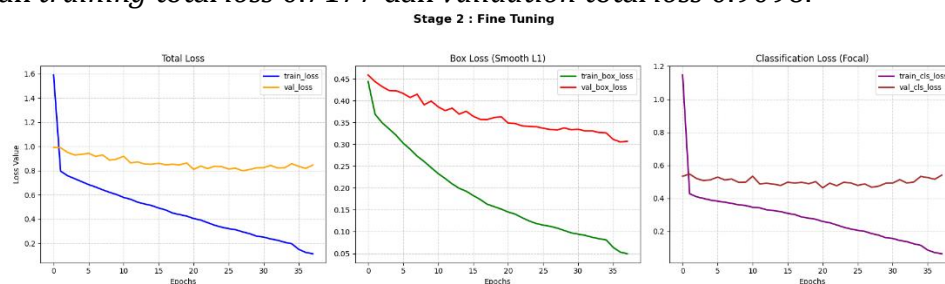


Gambar 4. Grafik *Loss Training* Tahap 1

Pada tahap pertama, seluruh lapisan pada backbone *Resnet50* dikunci atau dibekukan (*frozen*). Hal ini dilakukan agar pembaruan bobot hanya difokuskan pada penyesuaian lapisan *Feature Pyramid Network (FPN)* dan *Prediction Head*. Pelatihan tahap pertama ini dijalankan selama 20 *epoch*.

Grafik pada gambar 4, x menunjukkan tren pergerakan nilai kerugian (*loss*) yang sangat positif. Kurva *training loss* (garis biru/hijau/ungu) dan *validation loss* (garis oranye/merah/coklat) bergerak turun secara konsisten dan beriringan dari awal hingga akhir *epoch*. Penurunan yang selaras ini membuktikan bahwa jaringan kepala prediksi berhasil mempelajari fitur-fitur dasar dari objek X-ray tanpa mengalami indikasi *overfitting*.

Performa paling optimal pada tahap ini tercapai pada *Epoch* ke-18, dengan *training total loss* 0.7177 dan *validation total loss* 0.9096.



Gambar 5. Grafik *Loss Training* Tahap 2

Setelah jaringan *Prediction Head* stabil, tahap kedua dilanjutkan dengan membuka kembali status kunci pada seluruh lapisan backbone (*unfreeze*). Pada tahap *fine tuning* ini, seluruh bobot jaringan model dari ujung ke ujung diperbarui secara bersamaan menggunakan tingkat pembelajaran (*learning rate*) yang lebih kecil. Pelatihan tahap kedua ini dijadwalkan berjalan maksimal selama 40 *epoch*.

Pengamatan terhadap gambar 5, y memperlihatkan fenomena menarik pada awal proses pelatihan. Pada *Epoch* ke-1, metrik *training loss* mengalami lonjakan tiba-tiba (*spike*), di mana *total_loss* naik secara drastis menyentuh angka 1.5901 dan *classification_loss* mencapai 1.1463. Fenomena ini dikenal dengan istilah *gradient shock* (kejutan gradien). Hal ini sangat lumrah terjadi ketika jutaan parameter pada backbone yang sebelumnya berstatus beku mendadak "dibuka" dan langsung menerima aliran pembaruan bobot (*gradien*) dari lapisan prediksi.

Namun, model terbukti mampu memulihkan diri dengan sangat cepat. Mulai dari *epoch* kedua dan seterusnya, grafik kembali normal; nilai kerugian merosot tajam dan meneruskan tren penurunan yang stabil. Kurva *validation loss* juga perlahan ikut menurun dan mulai melandai (*plateau*) di pertengahan proses pelatihan. Performa terbaik pada tahap 2 *fine tuning* ini berhasil dicatat pada *Epoch* ke-28, dengan *training total loss* 0.2942 dan *validation total loss* 0.7985.

Evaluasi

Tabel 4. Evaluasi Keseluruhan Model

Metrik	Nilai	Persentase
<i>map (Precision)</i>	0.2337	23.37%
<i>Recall (mAR)</i>	0.3207	32.07%
<i>F1-Score</i>	0.2703	27.03%

Secara umum mencatatkan persentase nilai presisi (*map*) sebesar 23.37%,

tingkat sensitivitas deteksi (Recall) sebesar 32.07%, serta nilai keseimbangan di antara keduanya (F1-Score) sebesar 27.03%.

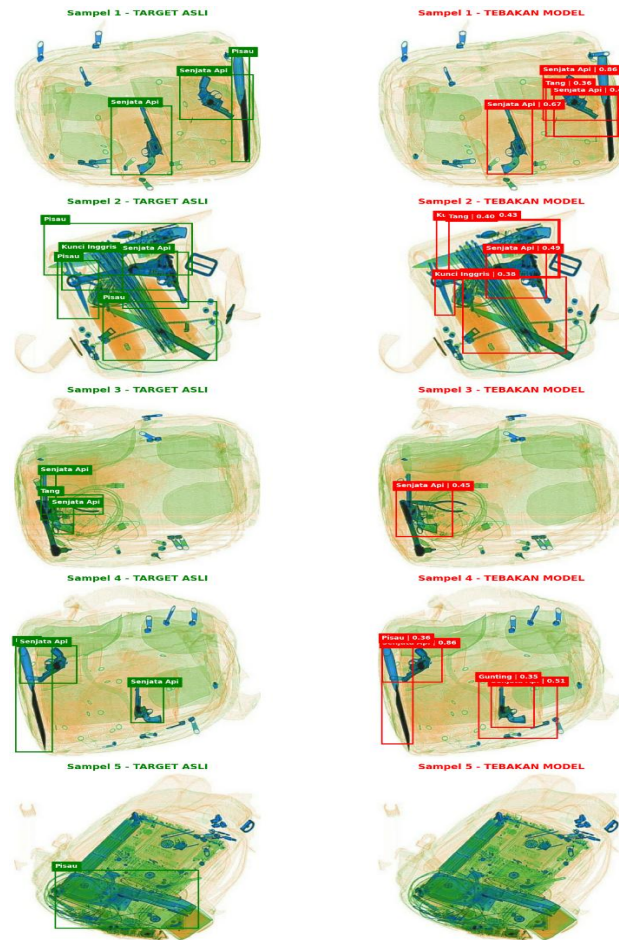
Untuk membedah performa akurasi secara lebih spesifik, evaluasi dilanjutkan menggunakan parameter pengujian COCO (Common Objects in Context). Metrik ini mengukur sensitivitas model terhadap toleransi pergeseran kotak pembatas (Intersection over Union/IoU) serta pengaruh ukuran objek terhadap tingkat keberhasilan deteksi.

Tabel 5. Evaluasi Berdasarkan Standar COCO

Parameter COCO	Nilai	Persentase
<i>map</i> @ IoU=0.50 (PASCAL VOC)	0.4371	43.71%
<i>map</i> @ IoU=0.75 (<i>Strict</i>)	0.2235	22.35%
<i>Recall</i> (Ancaman Kecil)	0.0143	1.43%
<i>Recall</i> (Ancaman Sedang)	0.2827	28.27%
<i>Recall</i> (Ancaman Besar)	0.3827	38.27%

Pengujian menggunakan parameter COCO menunjukkan bahwa model memperoleh nilai mAP @ IoU=0.50 sebesar 43.71% dan mAP @ IoU=0.75 sebesar 22.35%. Hasil ini menunjukkan bahwa model mampu melakukan deteksi objek ancaman pada citra X-Ray dengan tingkat akurasi yang cukup baik pada toleransi IoU menengah. Berdasarkan ukuran objek, performa terbaik diperoleh pada objek berukuran besar dengan *recall* sebesar 38.27%, sedangkan performa terendah terjadi pada objek kecil yang hanya mencapai *recall* sebesar 1.43%. Rendahnya kemampuan deteksi objek kecil dipengaruhi oleh beberapa faktor, seperti tingginya tingkat *occlusion* pada citra bagasi serta proses *resize* citra menjadi resolusi 416 × 416 yang menyebabkan detail objek kecil menjadi kurang representatif. Kondisi tersebut menyebabkan fitur visual objek kecil, seperti Pisau atau Gunting, sulit dipelajari secara optimal oleh model.

Pengujian



Gambar 6. Visualisasi *Ground Truth* vs Prediksi

Gambar 6 menunjukkan bahwa model telah mampu mengenali pola objek ancaman, khususnya pada objek berukuran besar seperti Senjata Api. Namun, hasil visualisasi masih memperlihatkan adanya beberapa *bounding box* ganda (*redundant bounding boxes*) yang saling bertumpukan pada objek yang sama. Kondisi ini menunjukkan bahwa algoritma *Non-Maximum Suppression* (NMS) telah bekerja dalam menyaring prediksi, tetapi parameter *threshold* yang digunakan, yaitu *confidence score* 0.35 dan *IoU* 0.5, masih belum cukup ketat untuk menghilangkan seluruh deteksi redundan. Selain itu, pada kondisi bagasi dengan tingkat *occlusion* tinggi, model hanya mampu mengenali sebagian fitur objek sehingga menghasilkan beberapa prediksi yang tumpang tindih. Oleh karena itu, penyesuaian *threshold IoU* pada NMS atau penggunaan *filtering* yang lebih ketat berpotensi meningkatkan kualitas hasil deteksi dan mengurangi kemunculan *bounding box* ganda.

KESIMPULAN

Berdasarkan hasil penelitian, model Resnet50 berbasis Convolutional Neural Network (CNN) berhasil diimplementasikan untuk mendeteksi objek berbahaya pada citra X-Ray bandara. Hasil evaluasi menunjukkan nilai Mean Average Precision (mAP) pada *IoU* 50% sebesar 43.71%, dengan performa terbaik pada objek berukuran besar seperti Senjata Api dan Tang yang mencapai recall sebesar 38.27%. Namun, model masih mengalami kesulitan dalam mendeteksi objek kecil dengan recall sebesar 1.43%, terutama akibat tingginya

occlusion dan berkurangnya detail visual setelah proses resize citra menjadi 416 × 416.

Secara visual, model telah mampu mengenali objek ancaman dengan cukup baik, meskipun masih ditemukan beberapa bounding box ganda pada hasil prediksi. Hal tersebut menunjukkan bahwa algoritma Non-Maximum Suppression (NMS) telah berjalan, tetapi masih memerlukan penyesuaian threshold filtering agar hasil deteksi menjadi lebih presisi. Secara keseluruhan, model yang dikembangkan menunjukkan performa yang cukup baik dan berpotensi digunakan sebagai pendukung sistem deteksi ancaman pada citra X-Ray bandara.

DAFTAR PUSTAKA

- Abu Ayub, G. (2021). Analisa Penggunaan Mesin X-Ray Sebagai Security System Di Bandara Internasional Husein Sastranegara. *Je-Unisla*, 6(2). www.jurnalteknik@unisla.ac.id/index.php/elektronika
- Aprianto, K., Mahdiyah, U., & Wulanningrum, R. (2025). Analisis Perbandingan Model PSO-LSTM dan LSTM Konvensional untuk Prediksi Harga Bitcoin di Market Cryptocurrency 1*. In *INOTEK* (Vol. 9).
- Awaludin, I., Fadhil, M., Andhika Zaini Zulfikor, M., Negeri Bandung Jl Gegerkalong Hilir, P., Parongpong, K., Bandung Barat, K., Barat, J., & Artikel, I. (2022). Analisis Kinerja ResNet-50 dalam Klasifikasi Penyakit pada Daun Kopi Robusta. *JURNAL INFORMATIKA*, 9(2). <http://ejournal.bsi.ac.id/ejurnal/index.php/ji>
- Firmansyah, Z., Asmarajati, D., Hidayat, M., Hasanah, N., Alif Muwafiq Baihaqy, M., Rohman, S., Sains Al-Qur, U., & Artikel, R. (2025). INFO ARTIKEL ABSTRAK. *TECHNOMEDIA: Informatics and Computer Science*, 2(2), 3047–2180. <https://doi.org/10.58641>
- Harahap, M., Laia, E. M., Sitanggang, L. S., Sinaga, M., Sihombing, D. F., & Husein, A. M. (2022). Detection of Covid-19 Disease in X-Ray Imagery With Convolutional Neural Network Approach (CNN). *Jurnal RESTI*, 6(1), 70–77. <https://doi.org/10.29207/resti.v6i1.3373>
- Hartato, B. P. (2021). Penerapan Convolutional Neural Network pada Citra Rontgen Paru-Paru untuk Deteksi SARS-CoV-2. *Jurnal RESTI*, 5(4), 747–759. <https://doi.org/10.29207/resti.v5i4.3153>
- Hastomo, W., & Satyo Bayangkari Karno, A. (2021). DIAGNOSA COVID-19 CHEST X-RAY DENGAN CONVOLUTION NEURAL NETWORK ARSITEKTUR RESNET-152. *Jurnal Riset Inovasi Bidang Informatika Dan Pendidikan Informatika (KERNEL)*, 2(1), 26–33.
- Herdianto, & Nasution, D. (2022). Prosiding SNASTIKOM: Seminar Nasional Teknologi Informasi & Komunikasi Paper Klasifikasi Objek Menggunakan Metode Convolutional Neural Network (CNN). *SNASTIKOM*.
- Jinan, A., & Hayadi, B. H. (2022). Klasifikasi Penyakit Tanaman Padi Menggunakan Metode Convolutional Neural Network Melalui Citra Daun (Multilayer Perceptron). In *Journal of Computer and Engineering Science* (Vol. 1, Number 2).
- Kusumadewi, P. M., Roisah, K., Prabandari, A. P., Ranaivo, R., & Manitra, M. (2023). ANALISIS STANDAR PENERBANGAN DI INDONESIA BERDASARKAN HUKUM UDARA INTERNASIONAL. *Jurnal Masalah-Masalah Hukum*, 52(3), 262–261.

- Liu, Y., Li, Q., Wang, K., Liu, J., He, R., Yuan, Y., & Zhang, H. (2021). Automatic Multi-Label ECG Classification with Category Imbalance and Cost-Sensitive Thresholding. *Biosensors*, *11*, 453. <https://doi.org/10.3390/bios>
- Marsya Aulia, D., & Dyahjatmayanti, D. (2024). Identifikasi Bahaya dan Penilaian Risiko di Area Apron Bandara Internasional Adi Soemarmo Solo. *HEMAT: Journal of Humanities Education Management Accounting and Transportation*, *1*(2), 605–617.
- Nasution, M. U., Harahap, L. S., & Syakbani, F. (2025). TINJAUAN METODE PENGOLAHAN CITRA DIGITAL UNTUK DETEKSI OBJEK OTOMATIS. *Jurnal Informatika Dan Teknik Elektro Terapan*, *13*(3). <https://doi.org/10.23960/jitet.v13i3.7135>
- Raghu, M., Unterthiner, T., Kornblith, S., Zhang, C., & Dosovitskiy, A. (2021). Do Vision Transformers See Like Convolutional Neural Networks? *NeurIPS (Neural Information Processing Systems)*.
- Rahbari, M., Rahlfs, S., Jortzik, E., Bogeski, I., & Becker, K. (2017). H2O2 dynamics in the malaria parasite *Plasmodium falciparum*. *PLoS ONE*, *12*(4). <https://doi.org/10.1371/journal>
- Rahmawati, A., Yulianti, I., Nurajizah, S., Hidayatulloh, T., & Oktarini Sari, A. (2025). Analisis Performa Model ResNet-50 Pada Diagnosis Pneumonia Balita Berdasarkan Citra Radiografi Thorax. *Computer Science (Co-Science)*, *5*(1), 42–48. <https://www.kaggle.com/datasets/paultimothymooney/chest>
- Sanjaya, J., & Ayub, M. (2020). Augmentasi Data Pengenalan Citra Mobil Menggunakan Pendekatan Random Crop, Rotate, dan Mixup. *Jurnal Teknik Informatika Dan Sistem Informasi*, *6*(2). <https://doi.org/10.28932/jutisi.v6i2.2688>
- Sumiyati, & Kifni Yudianto. (2024). Peran Pengawasan Petugas Aviation Security Dalam Mendeteksi Ancaman Barang Berbahaya Di Bandar Udara Rahadi Oesman Ketapang. *Jurnal Manajemen Riset Inovasi*, *2*(4), 145–155. <https://doi.org/10.55606/mri.v2i4.3180>
- Syafarina, G. A., Purnomo, I. I., & Hasbi, M. (2025). Klasifikasi Buah dan Sayuran Multi-Label Menggunakan CNN: Mengatasi Class Imbalance dengan Focal Loss. *Jurnal CoSciTech (Computer Science and Information Technology)*, *6*(3), 561–567. <https://doi.org/10.37859/coscitech.v6i3.10116>
- Tian, H., Zheng, Y., & Jin, Z. (2020). Improved RetinaNet model for the application of small target detection in the aerial images. *IOP Conference Series: Earth and Environmental Science*, *585*(1). <https://doi.org/10.1088/1755-1315/585/1/012142>
- Viriyasaranon, T., Chae, S. H., & Choi, J. H. (2022). MFA-net: Object detection for complex X-ray cargo and baggage security imagery. *PLoS ONE*, *17*(9 September). <https://doi.org/10.1371/journal.pone.0272961>
- Yang, X., Lan, T., & Xu, Y. (2025). A Novel Dangerous Goods Detection Network Based on Multi-Layer Attention Mechanism in X-Ray Baggage Images. *IEEE Access*, *13*, 106805–106816. <https://doi.org/10.1109/ACCESS.2025.3581450>
- Zhu, L., Lee, F., Cai, J., Yu, H., & Chen, Q. (2022). An Improved Feature Pyramid Network for Object Detection. <https://www.sciencedirect.com/science/article/pii/S0925231222001588>