BAB II
TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Beberapa penelitian terkait algoritma *deep learning* ataupun *machine learning* dalam menyelesaikan permasalahan sampah, mengatasi permasalahan *imbalanced dataset,* dan mengatasi permasalahan deteksi *small object* yang sulit untuk dideteksi telah dilakukan. Penelitian [40] memperkenalkan *TACO dataset* sebagai *novel-dataset* terhadap 60 kategori sampah yang masuk dalam *28 super-categories.* Selama proses pembelajaran dengan *Mask RCNN, dataset* di-*resize* ukurannya menjadi 1024x1024 piksel, *transplantasi* 320 *instances* yang di-*copy* pada gambar target, namun dilakukan pencampuran terlebih dahulu di level piksel dengan gambar target. Pencampuran tersebut dilakukan menggunakan *truncanted distance transform* pada *mask* biner sehingga menghasilkan *mask* yang lebih halus dan hasil komposisi tampak lebih alami. Selain itu, selama proses pembelajaran juga menerapkan *Gaussian blur* dan *noise AWS*, perubahan eksposur dan kontras gambar, rotasi gambar dalam rentang -45° hingga 45°, serta memotong area di sekitar *bounding boxes* agar dalam setiap potongan gambar selalu terlihat objek sampah. Penelitian ini dilatih hanya dengan menggunakan 10 *super categories* dan memperoleh AP 17.6% untuk 4-*fold cross validation.*

Penelitian [41] mengevaluasi performa *YOLOv4, YOLOv5*, dan *Faster RCNN* dalam mendeteksi sampah menggunakan *dataset TACO.* Penelitian tersebut dilakukan dalam dua fase. Fase pertama adalah hanya melakukan *auto-orientation* dan mengubah ukuran gambar menjadi 320x320, sedangkan fase kedua memperluas jangkauan *data augmentation*-nya dengan *auto-orientation, resize image* ke 640x640*,* menerapkan *RGB color space, mosaic augmentation,* dan *smoothing preprocessing*. Tidak hanya itu, penelitian tersebut juga hanya

menggunakan satu jenis kategori pada *dataset TACO,* yakni *litter* untuk mengatasi permasalahan *imbalanced.* Hasil terbaik yang diperoleh dari penelitian tersebut adalah mAP@0.5 88% dan mAP@0.5:0.95 71% oleh *YOLOv5.*

Penelitian [42] mengevaluasi performa *YOLOv8* dan *YOLOv9* dalam mendeteksi 60 kategori sampah pada *dataset TACO.* Penelitian tersebut menerapkan teknik augmentasi seperti *flipping horizontal* dan *vertical*, rotasi 90° ke segala arah, *cropping* dalam *range* 0%-15%, *shearing* secara *horizontal* dan *vertical,* transformasi 5% gambar ke *grayscale*, menyesesuaikan saturasi, warna, pencahayaan, kecerahan, memotong dan menambahkan blur menggunakan lima *boxes,* dengan ukuran masing-masing 2%. Hasil terbaik diperoleh dalam penelitian tersebut adalah 13.4% dan 9.8% untuk nilai mAP@50 dan mAP@50:95 oleh *YOLOv8*, sementara *YOLOv9* memperoleh nilai 3.29% dan 4.6% untuk nilai mAP@50:95 dan mAP@50.

Sementara itu penelitian yang dilakukan oleh [43] juga masih menggunakan *dataset TACO* untuk mengevaluasi performa *YOLOv8* dan *YOLOv5* dalam menerapkan deteksi sampah di lingkungan nyata pada *edge devices*. Ukuran gambar yang dipakai dalam melatih model dengan *YOLOv8* dan *YOLOv5* adalah 640x640, sedangkan untuk variasi *YOLOv5s* adalah 1280x1280. Teknik augmentasi yang digunakan adalah membalikkan gambar dengan probabilitas 0.5, memutarkan gambar secara acak dan menempelkannya ke gambar lain, sehingga menghasilkan *range degree* dari -10 hingga 10, dan *copy paste* objek untuk meningkatkan distribusi *instances.* Untuk mengoptimalkan ukuran model, penelitian tersebut melakukan kuantitasi *post-processing* dengan mengkonversi model ke dalam TFLite sebagai *floating point* setengah presisi, kuantitasi sebagai bilangan bulat 8-bit hanya dari bobot model, dan kuantitasi rentang dinamis. Hasil terbaik mAP@50 yang diperoleh adalah 68.6% oleh *YOLO-v5s6u*, mAP@50:95 terbaik yang diperoleh adalah 54.3% oleh *YOLO-v8s*, dan ukuran model paling optimal adalah 5 MB oleh *YOLO-v5n*.

Penelitian [44] mengombinasikan teknik SMOTE dan *transfer boosting* berbasis *instances* untuk menyeimbangkan kembali distribusi *skewed class.* Teknik *transfer boosting* yang digunakan dalam penelitian tersebut adalah *transfer Adaboost.* Penelitian tersebut melakukan duplikasi terhadap *instances,* melainkan menerapkan teknik *oversampling* SMOTE. Untuk menghasilkan sintesis objek, penelitian tersebut menerapkan dua pendekatan yakni (1) menghitung selisih antara vektor fitur yang dipertimbangkan dan salah satu dari *k* tetangga terdekatnya; (2) mengalikan selisih tersebut dengan bilangan acak antara 0 dan 1, lalu menambahkan hasilnya ke vektor fitur yang sedang dipertimbangkan. Hal tersebut akan mengakibatkan pemilihan titik acak pada garis yang menghubungkan kedua vektor fitur. Untuk menghemat komputasi pencarian *k-NN*, teknik yang digunakan dalam penelitian tersebut adalah *kd-tree*. Metode yang diusulkan dalam penelitian tersebut dibandingkan dengan algoritma *sampling-based, cost-sensitive algorithm,* dan *TrAdaboost-based algorithm.* Berdasarkan *sampling-based WSMOTE-TrAdaboost (ALL)* menghasilkan *instances sintesis* dan menyaring proporsi *instances* pelatihan sumber positif yang merugikan di setiap siklus *boosting* sehingga mampu meningkatkan kinerja pengklasifikasian. Namun, kinerjanya bergantung pada kinerja *Adaboost (ALL + SMOTE),* apabila kinerjanya lebih rendah daripada *Adaboost(TGT + SMOTE),* maka kinerja *WSMOTE-TrAdaboost (ALL)* cenderung menurun. Ditinjau berdasarkan *cost-sensitive algorithm,* metode *WSMOTE-TrAdaboost (ALL)* memperoleh kinerja terbaik dibandingkan *AdaC2.* Sementara itu, berdasarkan *TrAdaboost-based algorithm,* algoritma *WSMOTE-TrAdaboost (ALL)* cenderung bekerja lebih baik dibandingkan *TrAdaboost-based algorithm* pada kebanyakan kasus.

Penelitian [45] mengevaluasi performa SMOTE dengan *Tomek Link* dan ENN untuk menangani permasalahan *imbalanced dataset* yang diuji pada tiga belas *data sets* UCI, salah satu di antaranya adalah *dataset Satimage* yang merepresentasikan lingkungan piksel berukuran 3x3 dalam citra satelit. Penelitian dilakukan dengan menguji teknik *oversampling* dan *undersampling*, yakni *Random*

oversampling, Random under-sampling, Tomek Links, Condensed Nearest Neighbor Rule, One-sided selection, CNN + Tomek links, Neighborhood Cleaning Rule, Smote, Smote + ENN, dan *Smote + Tomek links.* Dalam menentukan nilai *k*-nya, penelitian tersebut menggunakan fungsi *Euclidean* untuk *distance*-nya pada *atribut kuantitatif,* dan VDM untuk atribut *kualitatif.* Dari penelitian tersebut diperoleh bahwa ukuran *dataset imbalanced* yang sedikit memengaruhi performa klasifikasi, karena kelas minoritas kurang terwakili oleh jumlah sampel yang sedikit. Selain itu juga, *Smote + ENN* dan *Smote + Tomek links* memiliki performa yang lebih baik untuk *imbalanced datasets* dengan jumlah sampel positif yang lebih sedikit. Sebaliknya, untuk *imbalanced datasets* dengan jumlah sampel positif yang lebih banyak, *Random Oversampling* memberikan hasil yang lebih baik.

Selanjutnya penelitian [26] menerapkan teknik *data resampling* untuk menangani *class imbalanced* pada deteksi objek *Insulator Defect Image Dataset (IDID).* Teknik *oversampling* dan *undersampling* dilakukan dengan membuat kembali *dataset 'D'* di mana setiap kelas *'i'* memiliki sejumlah *'K'* sampel. Pada *oversampling, 'K'* didefinisikan sebagai ukuran target sampel yang sama dengan ukuran kelas terbesar dalam *dataset*, sebaliknya untuk *undersampling, 'K'* didefinisikan sebagai ukuran target sampel yang sama dengan ukuran kelas terkecil dalam *dataset*. Model dilatih dengan YOLOv8 dengan mAP tertingginya 64%. Selain itu juga, kualitas dan representasi himpunan data mempengaruhi efektivitas teknik *resampling.*

Penelitian [46] berfokus mengusulkan *complement cross entropy,* sebuah *loss function* menetralkan probabilitas kelas yang salah tanpa ada prosedur tanpa prosedur pelatihan tambahan. Dengan demikian, dalam hal probabilitas *softmax* kelas *ground truth* lebih unggul dari kelas lain. Ketika kelas minoritas merupakan kelas *ground truth,* pendekatan ini mencegah kelas minoritas terancam oleh kelas-kelas lain termasuk kelas mayoritas selama proses pelatihan, sehingga memungkinkan model untuk menemukan pola tersembunyi dari kelas minorita. Model terbaik dilatih dengan *EfficientNet_B0 + CCE* memperoleh akurasi sebesar

99,8%. Namun, masih terdapat *misclasified* sampel berbeda yang terlihat sama yang oleh model.

Penelitian [34] mengusulkan ERF-RTMDet untuk meningkatkan *small objects* pada *Remote Sensing* (RS) *dataset*. Penelitian tersebut dilakukan dengan tiga modul, pertama menerapkan DSPPF untuk mengekstrak informasi kontekstual yang lebih banyak terhadap objek kecil. Kedua menerapkan *Hybrid Dilated Attention* (HDA) untuk mendapatkan informasi objek kecil yang lebih. Detail. Ketiga menerapkan CARAFE sebagai operator *upsampling,* memungkinkan *feature map* yang di-*upsampling* untuk memuat lebih banyak informasi tentang objek kecil. Objek kecil yang dilatih oleh RTMDet pada penelitian tersebut berhasil memperoleh mAPs 52%, namun memerlukan lebih banyak memori.

Penelitian [47] mengusulkan DS-YOLO, sebuah algoritma beebasis YOLOv8 untuk menangani deteksi *intelligent surveillance* pada *dataset CrowdHuman* dan *VisDrone2019* yang melibatkan sejumlah objek padat dan kecil. Penelitian tersebut memodifikasi struktur *backbone* YOLOv8 dengan C2fUIB yang lebih ringan dan memperluas bidang representatif. Tidak hanya itu, FPN-nya juga dimodifikasi dengan *Light-weight Full Scale PAFPN* (LFS-PAFPN) yang dikombinasikan dengan DO-C2F untuk mengurangi tingkat kesalahan pada objek padat kecil dengan tetap memastikan akurasi deteksi objek besar. Selanjutnya menerpakan teknik *DySample* untuk meminimalkan hilangnya fitur objek padat selama transmisi jaringan. Performa terbaik diperoleh dengan mAP@0.5 82,3% dan *params*-nya 8,288 untuk *dataset CrowdHuman* dan mAP@0.5 43,1 % dan *params*-nya 9,330 untuk *dataset VisDrone2019.* Meskipun demikian, waktu inferensi model memerlukan waktu yang lama, dan efektivitasnya terhadap berbagai kondisi cuaca dan pencahayaan belum ditinjau lebih lanjut.

Tabel 2. 1 Perbandingan hasil penelitian sebelumnya

No.	Fokus	Metode	Hasil	Keterbatasan	Ref
1.	Evaluasi performa *Mask RCNN* terhadap 10 *super categories dataset TACO.*	Menerapkan *Mask RCNN.* Teknik augmentasi yang digunakan adalah transplantasi 320 *instances* ke gambar target menggunakan *truncanted distance transform,* menerapkan *Gaussian blur* dan *noise AWS*, perubahan eksposur dan kontras gambar, rotasi gambar dalam rentang -45° hingga 45°, serta memotong area di sekitar *bounding boxes*	AP 17.6% untuk *4-fold cross validation*.	Hasil deteksi pada objek kecil masih buruk dan secara signifikan memengaruhi AP.	[40]
2.	Evaluasi performa *YOLOv4, YOLOv5,* dan *Faster RCNN* dalam deteksi sampah pada *dataset TACO.*	Menerapkan *YOLOv4, YOLOv5,* dan *Faster RCNN.* Teknik augmentasi yang digunakan a*uto-orientation, resize image* to 320x320 dan 640x640, *RGB color space, mosaic augmentation, smoothing preprocessing,* menggunakan	mAP@0.5 88% dan mAP@05:095 71% pada *YOLOv5.*	Nilai mAP yang tinggi hanya terbatas pada satu kategori, yakni *litter*.	[41]

No.	Fokus	Metode	Hasil	Keterbatasan	Ref
		label *litter* untuk menangani *imbalanced dataset.*			
3.	Evaluasi performa *YOLOv9* dan *YOLOv8* pada 60 kategori sampah di *dataset TACO.*	Menerapkan *YOLOv9* dan *YOLOv8*. Teknik augmentasi yang diterapkan *flipping horizontal* dan *vertical*, rotasi 90° ke segala arah, *cropping* dalam *range* 0%-15%, *shearing* secara *horizontal* dan *vertical,* transformasi 5% gambar ke *grayscale*, menyesesuaikan saturasi, warna, pencahayaan, kecerahan, memotong dan menambahkan blur menggunakan lima *boxes,* dengan ukuran masing-masing 2%.	Hasil terbaik mAP@50 dan mAP@50:95-nya adalah 13.4% dan 9.8% pada *YOLOv8,* sementara *YOLOv9* memperoleh nilai 3.29% dan 4.6% untuk nilai mAP@50:95 dan mAP@50.	Kinerja *YOLOv8* dan *YOLOv9* masih belum memadai.	[42]
4.	Mengevaluasi performa *YOLOv8* dan *YOLOv5* dalam menerapkan deteksi sampah dengan	Menerapkan *YOLO-v5n, YOLO-v5n6u, YOLO-v8n, YOLO-v5s, YOLO-v5s6u, YOLO-v8s*. Teknik augmentasi yang digunakan adalah membalikkan gambar dengan probabilitas	Hasil terbaik mAP@50 yang diperoleh adalah 68.6% oleh *YOLO-v5s6u*, mAP@50:95 terbaik yang diperoleh adalah	Terbatas hanya pada satu kategori, yakni *litter.*	[43]

No.	Fokus	Metode	Hasil	Keterbatasan	Ref
	dataset TACO di lingkungan nyata pada *edge devices.*	0.5, memutarkan gambar secara acak dan menempelkannya ke gambar lain, dan *copy paste* objek. Lalu, menerapkan kuantitasi *post-processing* dengan mengkonversi model ke dalam TFLite sebagai *floating point* setengah presisi, kuantitasi sebagai bilangan bulat 8-bit hanya dari bobot model, dan kuantitasi rentang dinamis.	54.3% oleh *YOLO-v8s*, dan ukuran model paling optimal adalah 5 MB oleh *YOLO-v5n*.		
5.	Mengatasi permasalahan *imbalanced class* pada *object recognition*	Menerapkan *WSMOTE-TrAdaboost(ALL)* dan *kd-tree* untuk menghemat ruang pencarian *k-NN.*	Menghasilkan *instances sintesis* dan menyaring proporsi *instances* pelatihan sumber positif yang merugikan di setiap siklus *boosting*.	Kurang efektif pada beberapa kategori objek *transfer learning*.	[44]
6.	Mengatasi permasalahan	*Tomek Links, Condensed Nearest Neighbor Rule, One-sided selection One-sided, CNN*	*Smote + ENN* dan *Smote + Tomek links* bekerja	Mengalosikan dataset dengan *sampel* kelas	[45]

No.	Fokus	Metode	Hasil	Keterbatasan	Ref
	imbalanced dataset pada tiga belas UCL *datasets*.	+ *Tomek links, Neighborhood Cleaning Rule, Smote, Smote + ENN,* dan *Smote + Tomek links*. *Euclidean Distance* untuk menentukan nilai *k* pada atribut kuantitatif, dan *VDM distance* untuk atribut kualitatif.	lebih baik pada *dataset* dengan sampel positif yang lebih sedikit. *Imbalanced datasets* dengan jumlah sampel positif yang lebih banyak, *Random Oversampling* memberikan hasil yang lebih baik.	positif mayoritas ke kelas positif minoritas belum tentu menghasilkan performa yang lebih baik.	
7.	Mengevaluasi efektivitas teknik *resampling* untuk menangani *class imbalanced* pada deteksi objek IDID.	Teknik *oversampling* dan *undersampling*. Pelatihan model dilakukan menggunakan YOLOv8.	Model dilatih dengan YOLOv8 dengan mAP tertingginya 64%.	Kualitas dan representasi himpunan data mempengaruhi efektivitas teknik *resampling,* sehingga ada perlakuan berbeda jika diterapkan pada dataset dengan distribusi yang berbeda.	[26]

No.	Fokus	Metode	Hasil	Keterbatasan	Ref
				Jika himpunan data tidak menangkap distribusi objek di dunia nyata secara akurat, hasilnya mungkin tidak dapat digeneralisasi dengan baik	
8.	Menerapkan sebuah *loss function* yang mampu memudahkan model mempelajari informasi penting dari sampel minoritas.	*Complement Cross Entropy Loss Function* dan klasifikasi dengan *ResNet-34*, *SqueezeNet*, dan *EfficientNet*.	Model terbaik dilatih dengan *EfficientNet_B0 + CCE* memperoleh akurasi sebesar 99,8%.	Terjadi *misclassified* oleh model pada sampel yang terlihat sama.	[46]
9.	Meningkatkan performa deteksi objek kecil pada *Remote Sensing datasets.*	DSPFF, HDA, CARAFE, dan RTMDet.	Berhasil mencapai 52% pada mAPs-nya.	Memerlukan memori tambahan.	[34]

No.	Fokus	Metode	Hasil	Keterbatasan	Ref
10.	Meningkatkan performa untuk deteksi objek kecil dan padat.	Struktur *backbone* YOLOv8 diubah dengan C2fUIB yang lebih ringan dan memperluas bidang representatif. Tidak hanya itu, FPN-nya juga dimodifikasi dengan *Light-weight Full Scale PAFPN* (LFS-PAFPN) yang dikombinasikan dengan DO-C2F.	Performa terbaik diperoleh dengan mAP@0.5 82,3% dan *params*-nya 8,288 untuk *dataset CrowdHuman* dan mAP@0.5 43,1 % dan *params*-nya 9,330 untuk *dataset VisDrone2019.*	Waktu inferensi model memerlukan waktu yang lama, dan efektivitasnya terhadap berbagai kondisi cuaca dan pencahayaan belum ditinjau	[47]

2.2 Kebaruan Penelitian

Tabel 1 merupakan bukti perkembangan penerapan teknologi dalam kehidupan sehari-hari secara signifikan. Dapat dilihat bahwa penelitian terkait dengan penerapan *Mask-RCNN* [40], *YOLOv4, YOLOv5,* dan *Faster RCNN* [41], *YOLOv9* dan *YOLOv8* [42], *YOLOv8* dan *YOLOv5* [43] terhadap deteksi sampah dengan *dataset* TACO telah dilakukan sebelumnya. Akan tetapi, hasil yang diberikan belum begitu baik, dan masih terdapat kendala ketidakseimbangan data antar kelas dan keterbatasan ruang fitur yang menyebabkan model lebih bias terhadap *large* ataupun *medium object,* dan mengabaikan *small object.* Dalam mengatasi ketidakseimbangan data untuk deteksi objek sampah tersebut, penelitian yang telah dilakukan sebelumnya hanya berfokus dengan menggabungkan berbagai macam kategori menjadi satu kategori sampah [41], [43], dan menggunakan 10 kategori sampah [33]. Adapun penelitian yang menggunakan 60 kategori belum memberikan performa yang lebih baik [35].

Tidak hanya itu, penelitian lain untuk mengatasi permasalahan ketidakseimbangan data pun telah dilakukan dengan menggunakan teknik *WSMOTE-TrAdaboost (ALL)* pada *object recognition* [44], *Tomek Links, Condensed Nearest Neighbor Rule, One-sided selection One-sided, CNN + Tomek links, Neighborhood Cleaning Rule, Smote, Smote + ENN,* dan *Smote + Tomek links*[45] terhadap data tabular*,* teknik *oversampling* dan *undersampling* pada deteksi objek [26], dan menerapkan *Complement Cross Entropy Loss Function* sebagai fungsi *loss* pada klasifikasi gambar [46]. Selain itu juga, penelitian untuk meningkatkan performa deteksi terhadap *small object* juga telah dilakukan dengan menerapkan CARAFE sebagai operator *upsampling*-nya [34], dan memperluas jangkuan *feature maps* pada FPN-nya dengan menerapkan *Light-weight Full Scale PAFPN* (LFS-PAFPN) untuk menangkap detail *small object* [47].

Meskipun deteksi objek sampah telah banyak dilakukan, namun model masih mengalami kendala dalam permasalahan *imbalanced dataset, scaled imbalanced,*

dan *small object detection* pada *background* yang kompleks. Selain itu juga, teknik *oversampling* dan *undersampling* untuk mengatasi *imbalanced dataset* masih jarang diterapkan dalam deteksi objek, lebih cenderung digunakan dalam tugas klasifikasi gambar [46], *object recognition* [44], dan data tabular [45]. Meskipun ada penelitian yang mengeskplor *oversampling* dan *undersampling* untuk deteksi objek, namun apabila dilakukan pada dataset dengan distribusi yang berbeda akan mempengaruhi efektivitas teknik *resampling* [26]. Selain itu juga, dengan menerapkan *Complement Cross Entropy Loss Function* masih terjadi *misclassified* terhadap sampel yang terlihat sama [46]. Tidak hanya itu, meskipun CARAFE dapat meningkatkan deteksi terhadap objek kecil, namun diperlukan memori yang lebih besar dalam komputasinya [34], [35]. Berdasarkan uraian tersebut masih ditemukan celah dari penelitian yang telah dilakukan sebelumnya untuk dieksplor lebih dalam lagi, khususnya dalam menangani *class imbalance,* dan deteksi terhadap *small object* pada *background* kompleks yang sering terjadi pada objek sampah. Dengan demikian untuk mengisi celah tersebut, penelitian ini memiliki kontribusi sebagai berikut:

1) Pertama, menerapkan teknik *resampling, repeat factor sampling* yang berpotensi untuk mengatasi *class imbalance* pada deteksi sampah, sehingga mampu meningkatkan kemampuan model dalam deteksi objek dari kelas minoritas tanpa memerlukan perubahan arsitektur model utama maupun penambahan data citra baru.
2) Kedua, menerapkan RTMDet sebagai operator deteksi dan klasifikasi objek. RTMDet digunakan karena kemampuannya yang baik dalam mendeteksi *small object* [48], dan memiliki *trade-off* yang baik antara akurasi dan kecepatan [21].
3) Ketiga, menerapkan *DySample* sebagai operator *upsampling* pada FPN. *DySample* digunakan karena lebih fokus pada wilayah target dan melakukan reduksi *noise* pada latar belakang dibanding teknik *interpolasi,* kemampuannya yang secara adaptif menyesesuaikan bobot *upsampling*

berdasarkan fitur *input.* Tidak hanya itu, *DySample* dapat menjaga biaya komputasi tetap rendah dibandingkan metode *dynamic upsampling* berbasis kernel seperti CARAFE [35].

4) Keempat, menerapkan *Efficient Multi-Scale Attention* (EMA) pada *feature map* hasil proses *feature fusion* di bagian *neck*, tepatnya pada P4 sebagai *feature refinement* sebelum *feature map* dipakai oleh *detection head* untuk prediksi *bounding box* dan klasifikasi objek. RTMDet telah memiliki kemampuan ekstraksi fitur dan fusi fitur multiskala yang baik, tetapi pada objek sampah dengan karakteristik visual yang beragam masih terdapat peluang untuk memperkuat representasi fitur sebelum proses prediksi. Oleh karena itu, EMA ditempatkan pada P4 sebelum *convolution head* untuk mempertahankan informasi setiap *channel* dan memberikan bobot perhatian pada fitur yang lebih relevan
5) Kelima, menerapkan *Ghost Conv module* sebagai ganti *Conv module* pada CSP-*Layer* di *backbone* tepatnya pada *stage layer* 4 untuk mengurangi kompleksitas parameter dengan memanfaatkan redudansi fitur dan mengoperasikannya dengan biaya yang hemat [22].

Secara keseluruhan, penelitian ini melakukan pendekatan yang efisien dan fleksibel dengan menerapkan teknik *resampling, DySample, Efficient Multi-Scale Attention* (EMA), *Ghost Conv module* pada RTMDet sebagai metode untuk menangani permasalahan deteksi objek sampah *outdoor*, dengan berfokus pada permasalahan ketidakseimbangan data antar kelas, peningkatan, performa deteksi objek kecil, representasi fitur multi-skala, serta pengurangan kompleksitas model.