



Artikel Penelitian

Perancangan Sistem Identifikasi Jenis Sampah Menggunakan Tensorflow Object Detection Dan Transfer Learning

Aulia Anshari Fathurrahman^a, Fajril Akbar^{a}**"Departemen Sistem Informasi Universitas Andalas, Padang, 25126, Indonesia"*

INFORMASI ARTIKEL

Sejarah Artikel:

Diterima Redaksi: 16 Agustus 2023

Revisi Akhir: 12 Mei 2024

Diterbitkan Online: 15 Mei 2024

KATA KUNCI

Identifikasi jenis sampah,

*transfer learning,**TensorFlow Object Detection*

KORESPONDENSI

E-mail: ijab@it.unand.ac.id

A B S T R A C T

Penanganan sampah menjadi tantangan utama dalam menjaga kebersihan dan kelestarian lingkungan. Identifikasi jenis sampah secara efisien sangat penting untuk meningkatkan pemilahan dan pengelolaan sampah. Penelitian ini berfokus pada penerapan teknologi deep learning menggunakan TensorFlow Object Detection API dan transfer learning untuk mengidentifikasi jenis sampah berdasarkan citra. Tujuan penelitian adalah mengembangkan model deep learning yang mampu mengidentifikasi sampah umum di Indonesia, serta mengevaluasi kinerjanya. Penelitian ini menggunakan TensorFlow Object Detection API dan transfer learning untuk melatih model identifikasi sampah dengan dataset yang telah dikumpulkan. Hasil penelitian menunjukkan tingkat akurasi yang tinggi dalam mengenali sampah seperti kaca, karton, kertas, logam, dan plastik. Dalam pengujian, model mencapai nilai Mean Average Precision (mAP) sebesar 0,858 dan Average Recall (AR) sebesar 0,91 pada data uji, dengan menggunakan nilai Intersection over Union (IoU) sebesar 0,841. Hasil ini menunjukkan kemampuan yang baik dalam mengenali dan membedakan jenis sampah. Penelitian ini berfokus pada Tempat Pengelolaan Sampah Nagari Tanjuang Gadang, yang mengalami peningkatan produksi sampah akibat pertumbuhan populasi dan urbanisasi. Infrastruktur yang tidak memadai dan pemilahan sampah manual yang masih dominan menimbulkan masalah lingkungan dan kesehatan. Dengan mengaplikasikan teknologi deep learning, penelitian ini mengusulkan solusi untuk meningkatkan efisiensi pengelolaan sampah dan mengurangi dampak negatif. Hasil penelitian ini diharapkan menjadi landasan bagi pengembangan sistem identifikasi jenis sampah yang lebih luas dan dapat diimplementasikan di berbagai fasilitas pengelolaan sampah, termasuk tempat pemilahan sampah di komunitas.

1. PENDAHULUAN

Pengelolaan sampah merupakan salah satu masalah utama di seluruh dunia, terutama di negara-negara berkembang. Perkembangan ekonomi dunia telah meningkatkan taraf hidup masyarakat, dan produksi sampah rumah tangga semakin meningkat dari tahun ke tahun. Saat ini, dunia menghasilkan 2,01 miliar ton sampah kota setiap tahun, yang merupakan kerusakan besar bagi lingkungan ekologis. Produksi sampah akan meningkat 70% jika kondisi saat ini terus berlanjut[1]. Pengelolaan sampah di Indonesia menjadi masalah yang semakin mendesak, terutama di kota-kota besar. Berdasarkan data Kementerian Lingkungan Hidup dan Kehutanan Republik

Indonesia, rata-rata produksi sampah harian sekitar 136.000 ton per hari pada tahun 2020 di Indonesia. Ini terus meningkat seiring dengan pertumbuhan penduduk dan kurangnya infrastruktur .

Tempat Pengelolaan Sampah Nagari Tanjuang Gadang memiliki peran penting dalam mengatasi masalah pengelolaan sampah di wilayah tersebut. Namun, pertumbuhan populasi dan urbanisasi yang pesat telah menyebabkan peningkatan signifikan dalam produksi sampah. Infrastruktur pengelolaan sampah yang ada tidak dapat menangani lonjakan volume sampah ini, yang menghasilkan berbagai permasalahan. Pengelolaan sampah yang tidak efektif menjadi salah satu tantangan utama, berpotensi menciptakan pencemaran lingkungan dan dampak negatif pada kesehatan masyarakat. Selain itu, pemilahan sampah secara

manual yang masih dominan memerlukan banyak pekerja, berisiko terhadap kesehatan, dan memerlukan waktu yang lama [2].

Pengelolaan sampah atau bahan limbah yang tidak efektif dapat menyebabkan pencemaran lingkungan dan kesehatan masyarakat. Banyak bahan limbah yang muncul sebagai sisa dari produk yang digunakan dalam kehidupan kita sehari-hari. Beberapa bahan limbah menciptakan efek toksik selama bertahun-tahun dan mencapai tubuh manusia secara berbeda dan menjadi berbahaya bagi kesehatan manusia. Untuk mengurangi bahaya produk-produk ini terhadap kesehatan, tindakan pengurangan bahaya harus dilakukan dengan mengumpulkannya kembali. Bisnis sampah dan daur ulang sedang kewalahan oleh volume sampah global yang terus meningkat [3].

Daur ulang dengan cepat menjadi komponen penting dari masyarakat yang berkelanjutan. Namun, seluruh proses daur ulang memiliki biaya tersembunyi yang tinggi [4]. Hal ini disebabkan oleh pemilihan, pemisahan, dan pengolahan bahan daur ulang. Meskipun banyak orang saat ini dapat melakukan pemilahan sampah mereka sendiri, mereka mungkin bingung tentang bagaimana memilih kategori sampah yang benar ketika membuang berbagai macam barang, menemukan pendekatan otomatis dan berbasis informasi untuk daur ulang saat ini sangat berharga di dunia industri karena menawarkan manfaat lingkungan dan ekonomi[5].

Kehidupan manusia dan lingkungan sama-sama dipengaruhi oleh pembuangan limbah, baik secara langsung maupun tidak langsung. Konsekuensi negatif dari bahan limbah dapat dikurangi dengan penggunaan sistem pengelolaan limbah yang kompeten [6]. Masalah lingkungan meningkat dalam jangka panjang jika sampah tidak ditangani dengan baik. Hal ini juga menjadi penghambat pembangunan berkelanjutan[7]. Beberapa komponen bekas dapat didaur ulang. Daur ulang tidak hanya dapat menghemat atau mengurangi emisi, tetapi juga menciptakan nilai dan mengurangi biaya[8]. Saat ini, ada dua jenis klasifikasi dan pemisahan sampah: klasifikasi sampah manual dan klasifikasi sampah otomatis menggunakan beberapa teknik. Yang pertama dapat dicapai dengan kecerdasan dan kekuatan manusia, sedangkan yang kedua memerlukan pencarian otomatis untuk teknik klasifikasi limbah yang sesuai [9]. Pemilahan sampah secara manual sering kali dilakukan oleh pekerja di tempat-tempat pemilahan sampah atau fasilitas daur ulang. Meskipun metode ini telah lama digunakan dan berhasil dalam memilah beberapa jenis sampah, namun memiliki beberapa kelemahan[10].

Sebagai solusi untuk mengatasi kendala pemilahan sampah manual, klasifikasi sampah otomatis menggunakan teknologi *deep learning* menjadi pilihan yang menarik. Teknologi ini memanfaatkan kecerdasan buatan dan algoritma pembelajaran mesin yang mampu mengenali dan membedakan berbagai jenis sampah berdasarkan citra atau data sensor lainnya [5], [10], [11], [12], [13]. Dengan menggunakan teknologi *deep learning*, sistem dapat dilatih untuk mengidentifikasi dan mengklasifikasikan sampah secara otomatis dengan akurasi yang tinggi. Teknologi *deep learning* ini telah banyak digunakan dalam berbagai bidang seperti pengolahan citra dan pengenalan pola [14], [15], [16], [17]. Penerapan teknologi *deep learning* untuk mendeteksi objek

berdasarkan citra juga dapat dilakukan dengan menggunakan *framework TensorFlow Object Detection API* dan *transfer learning*. Dalam penelitian yang dilakukan oleh [18]. Penggunaan teknologi *deep learning* dalam mengidentifikasi jenis sampah berdasarkan citra mampu mencapai akurasi sebesar 97,63%. Metode identifikasi jenis sampah telah dikembangkan menggunakan berbagai teknik, termasuk pengolahan citra dan analisis fitur. Identifikasi jenis sampah telah didefinisikan sebagai "proses untuk mengelompokkan sampah berdasarkan atribut-atribut tertentu, seperti jenis material, tekstur, warna, atau bentuk, dengan menggunakan metode pemrosesan citra atau analisis fitur"[13].

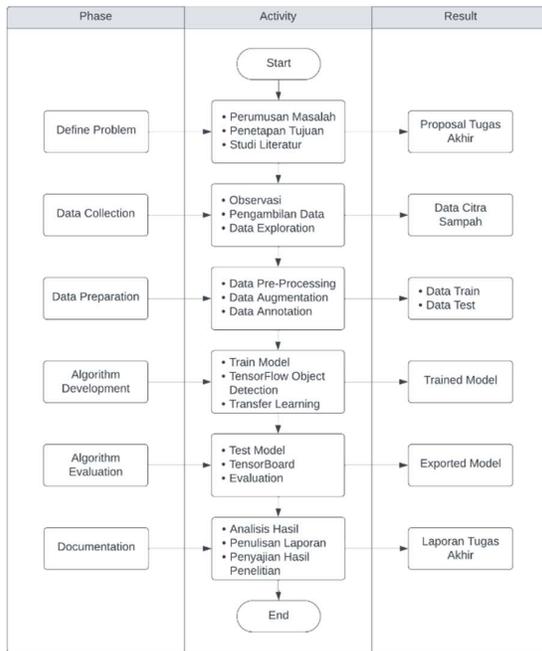
Namun, di Indonesia, penelitian tentang penggunaan teknologi *deep learning* untuk identifikasi jenis sampah berdasarkan citra masih terbatas. Oleh karena itu, penelitian ini bertujuan untuk mengembangkan model *deep learning* untuk identifikasi jenis sampah berdasarkan citra dengan menggunakan *TensorFlow Object Detection API* dan *transfer learning*. Diharapkan penelitian ini dapat memberikan kontribusi pada pengembangan teknologi pengelolaan sampah yang lebih efisien dan berkelanjutan dan mengujinya pada sampah yang umum di Indonesia seperti sampah organik dan non-organik.

Berdasarkan latar belakang yang telah dijelaskan sebelumnya, penelitian ini bertujuan untuk mengkaji dan mengembangkan metode deteksi dan klasifikasi sampah berdasarkan citra menggunakan teknologi *Deep Learning*. Penelitian ini akan fokus pada penerapan teknologi *Deep Learning* dalam mengidentifikasi jenis sampah secara otomatis berdasarkan citra yang diambil.

2. METODE

Objek penelitian ini adalah jenis sampah umum yang ditemukan di wilayah Tempat Pengelolaan Sampah Nagari Tanjung Gadang, Kec. Lareh Sago Halaban, Kab. Lima Puluh Kota. Objek sampah mencakup lima jenis sampah utama, yaitu kaca, karton, kertas, logam, dan plastik. Peneliti juga mengambil sampel sampah dari lokasi penelitian dan daerah sekitarnya untuk digunakan sebagai data pelatihan dan pengujian dalam pengembangan model identifikasi. Peneliti melakukan identifikasi dan klasifikasi jenis sampah ini menerapkan teknologi *deep learning* dengan *TensorFlow Object Detection API* dan *transfer learning*.

Pada penelitian ini, metode pendekatan yang digunakan adalah penelitian eksperimental. Penelitian eksperimental dilakukan dengan melakukan pengujian dan evaluasi terhadap performa model. Desain penelitian yang dilakukan dapat dilihat pada gambar 1.



Gambar 1. Diagram alur penelitian

Pada tahap pengujian, dilakukan pengujian model yang telah dilatih menggunakan *dataset* yang belum pernah dilihat sebelumnya. Tujuan pengujian ini adalah untuk mengevaluasi kemampuan generalisasi model dalam mengenali dan membedakan jenis sampah berdasarkan citra pada situasi yang realistis. Berikut ini adalah langkah-langkah yang dilakukan dalam pengujian model:

2.1. Pengujian Model pada Dataset yang Belum Dilihat Sebelumnya

Dataset yang digunakan untuk pengujian harus berbeda dengan *dataset* yang digunakan saat pelatihan dan validasi model. *Dataset* ini harus mencakup citra-citra sampah yang beragam dan mencerminkan kondisi yang sebenarnya. Pengujian dilakukan dengan menjalankan model pada *dataset* pengujian dan mencatat hasil prediksi yang diberikan oleh model. Hasil ini kemudian akan digunakan untuk evaluasi performa model.

2.1.1. Pelatihan Model

Pada sub-bab ini, akan dijelaskan mengenai proses pelatihan model *deep learning* menggunakan data *training* yang telah disiapkan. Pelatihan model merupakan tahap krusial dalam pengembangan sistem identifikasi jenis sampah berbasis *deep learning*. Setelah data *training* telah dipersiapkan dan dikonversi ke format *TFRRecords*, langkah selanjutnya adalah melatih model *deep learning* menggunakan *TensorFlow Object Detection API*. Pelatihan model bertujuan untuk mengoptimalkan parameter dan bobot model agar dapat mengenali dan mengklasifikasikan sampah dengan akurasi yang tinggi. Berikut adalah langkah-langkah yang perlu dilakukan:

1. Pemilihan Model Awal (*Pre-trained Model*)

Dalam pengembangan sistem identifikasi jenis sampah berbasis *deep learning*, pemilihan model awal yang tepat sangat penting. *Pre-trained model* yang telah dilatih pada *dataset* yang luas dapat memberikan keunggulan dalam

pelatihan model identifikasi jenis sampah. Model awal yang dipilih dan digunakan dalam penelitian ini adalah *SSD MobileNet V2 FPNLite 640x640*. Model tersebut diunduh dari repositori *TensorFlow Model Zoo* yang telah dilatih menggunakan *COCO 2017 Dataset*. Pemilihan model ini mempertimbangkan kecepatan inferensi model, keringanan model dan *mAP* yang cukup baik.

2. Transfer Learning dan Fine-tuning

Setelah memilih *pre-trained model* yang sesuai sebagai model awal, langkah selanjutnya adalah melakukan *transfer learning* dan *fine-tuning* untuk melatih model identifikasi jenis sampah. menggunakan *pre-trained model* sebagai dasar untuk *transfer learning*. Lapisan-lapisan awal dari *pre-trained model* telah dilatih untuk mengenali fitur-fitur umum pada *dataset* yang luas. Lapisan-lapisan awal tersebut tetap menggunakan bobot awalnya (*frozen*), sehingga informasi yang telah dipelajari tidak hilang. Setelah itu, menambahkan lapisan-lapisan akhir pada *pre-trained model* sesuai dengan jumlah dan jenis kelas sampah yang ingin diidentifikasi. Lapisan-lapisan ini akan belajar mengenali fitur-fitur spesifik pada *dataset* jenis sampah. Selanjutnya, melakukan *fine-tuning* pada model dengan melatihnya menggunakan data *training* yang telah disiapkan. Biarkan lapisan-lapisan awal tetap "beku" (*frozen*), sehingga bobot mereka tidak berubah. Lapisan-lapisan baru yang ditambahkan akan disesuaikan dan dioptimalkan untuk tugas identifikasi jenis sampah. Selama proses *fine-tuning*, perlu dilakukan monitor dan evaluasi kinerja model secara berkala. Memantau metrik evaluasi seperti akurasi, *loss*, dan metrik lain yang relevan. Dan melakukan penyesuaian pada konfigurasi dan *hyperparameter* untuk meningkatkan performa model. Pengaturan konfigurasi dan *hyperparameter* akan dijelaskan pada sub-bab selanjutnya.

3. Konfigurasi Pipeline dan Hyperparameter

Dalam pelatihan model identifikasi jenis sampah, konfigurasi *pipeline* dan pengaturan *hyperparameter* dapat mempengaruhi performa dan keberhasilan model. Berikut adalah langkah-langkah yang perlu dilakukan dalam konfigurasi *pipeline* dan pengaturan *hyperparameter*:

1. Konfigurasi Pipeline:

- Arsitektur model yang akan digunakan adalah *SSD*. Dan memindahkan *pipeline.config pre-trained model* ke dalam penyimpanan *checkpoint*.
- Memilih jumlah dan tipe lapisan yang akan digunakan dalam model, termasuk lapisan konvolusi, *pooling*, dan lapisan klasifikasi. Dalam hal ini ditambahkan 4 layer baru dan diakhiri dengan layer akhir sesuai jumlah kelas yang diidentifikasi.
- Mengatur ukuran *input* gambar yang akan digunakan dalam pelatihan. Dalam hal ini ukuran *input* yang digunakan adalah 640x640 piksel.
- Mendefinisikan metode augmentasi data seperti *flipping*, dan pergeseran untuk memperkaya *dataset training*.

2. Hyperparameter:

- Menentukan *hyperparameter* yang akan digunakan dalam pelatihan model.

b. Beberapa *hyperparameter* yang perlu diperhatikan antara lain:

1. *Learning rate*: Parameter yang mengontrol sejauh mana bobot model diperbarui pada setiap langkah pelatihan. *Learning rate* menggunakan *decay* dengan nilai awal 0.026666, penggunaan *learning rate decay* akan menyesuaikan nilai sesuai dengan perubahan *loss training*.
 2. *Batch size*: Jumlah sampel yang diproses dalam satu iterasi pelatihan. Jumlah *batch size* yang digunakan adalah 8, mengingat kapasitas komputasi *gpu* yang tersedia.
 3. *Epochs*: Jumlah iterasi yang akan dilakukan dalam pelatihan. Jumlah iterasi yang akan digunakan adalah 2000 dan akan ditingkatkan menjadi 4000 jika hasil *training* belum cukup baik.
 4. *Optimizer*: Algoritma optimasi yang digunakan untuk memperbarui bobot model. *Optimizer* yang digunakan adalah *Adam (Adaptive Moment Estimation)*.
 5. *Regularization*: Teknik yang digunakan untuk mencegah overfitting, seperti L1 atau L2 *regularization*.
 6. *Dropout*: Teknik yang mengurangi *overfitting* dengan secara acak mengabaikan sebagian *node* dalam jaringan.
3. Penyimpanan Model.
Menentukan tempat penyimpanan *checkpoint* model selama proses pelatihan. *Pipeline.config Pre-trained model* pada awalnya belum bisa dipakai sehingga perlu dilakukan perubahan. Untuk mengubah *file pipeline.config* agar sesuai dengan konfigurasi yang diinginkan dengan menggunakan *library text format*. Dengan melakukan konfigurasi *pipeline* yang tepat dan pengaturan *hyperparameter* yang sesuai, agar dapat memaksimalkan performa model identifikasi jenis sampah. Selanjutnya, dapat dilakukan proses pelatihan model.
4. Proses Pelatihan Model. Setelah melakukan konfigurasi *pipeline* dan pengaturan *hyperparameter*, langkah selanjutnya adalah menjalankan proses pelatihan model identifikasi jenis sampah. Proses pelatihan dilakukan dengan menjalankan *script* yang disediakan oleh *TensorFlow Object Detection API*. Yaitu dengan menjalankan *file "model_main_tf2.py"* yang ada dalam direktori "research/object_detection". Dengan menggunakan *argument path pipeline.config* untuk konfigurasi model dan *num_train_steps* untuk jumlah iterasi pelatihan. Setelah *script* pelatihan model dijalankan, proses pelatihan menampilkan hasil pelatihan untuk setiap 100 iterasi dan menyimpan bobot pelatihan terbaik pada direktori *checkpoint* setiap 500 iterasi. Proses pelatihan berhenti pada iterasi 2000 dan menghasilkan *loss* sebesar 0.24486871 dengan *learning rate* akhir sebesar 0.79917181 dalam waktu 13 menit 55 detik. Dapat dilihat pada Gambar 2.

```
INFO:tensorflow:Step 1900 per-step time 0.361s
10619 07:31:15.655900 140581650736900 model_lib_v2.py:705] Step 1900 per-step time 0.361s
INFO:tensorflow:{'loss/classification_loss': 0.07728567,
'loss/localization_loss': 0.015125946,
'loss/regularization_loss': 0.14503178,
'loss/total_loss': 0.23744339,
'learning_rate': 0.07993342}
10619 07:31:15.656373 140581650736900 model_lib_v2.py:708] {'loss/classification_loss': 0.07728567,
'loss/localization_loss': 0.015125946,
'loss/regularization_loss': 0.14503178,
'loss/total_loss': 0.23744339,
'learning_rate': 0.07993342}
INFO:tensorflow:Step 2000 per-step time 0.364s
10619 07:31:15.658300 140581650736900 model_lib_v2.py:705] Step 2000 per-step time 0.364s
INFO:tensorflow:{'loss/classification_loss': 0.00741699,
'loss/localization_loss': 0.013204906,
'loss/regularization_loss': 0.14424602,
'loss/total_loss': 0.24486871,
'learning_rate': 0.07991781}
10619 07:31:15.658791 140581650736900 model_lib_v2.py:708] {'loss/classification_loss': 0.00741699,
'loss/localization_loss': 0.013204906,
'loss/regularization_loss': 0.14424602,
'loss/total_loss': 0.24486871,
'learning_rate': 0.07991781}
```

Gambar 2. Hasil *Training* dalam 2000 iterasi

Dalam proses pelatihan menggunakan 2000 iterasi ditemukan hasil belum cukup baik, peneliti melanjutkan proses pelatihan untuk 4000 iterasi. Sehingga menghasilkan *loss* sebesar 0.1868276 dengan *learning rate* akhir sebesar 0.7926236 dalam waktu 13 menit 40 detik. Dapat dilihat pada Gambar 3.

```
INFO:tensorflow:Step 3800 per-step time 0.360s
10619 07:55:15.487631 139773631145792 model_lib_v2.py:705] Step 3800 per-step time 0.360s
INFO:tensorflow:{'loss/classification_loss': 0.0523538,
'loss/localization_loss': 0.009429479,
'loss/regularization_loss': 0.1346481,
'loss/total_loss': 0.19643137,
'learning_rate': 0.079357104}
10619 07:55:15.487903 139773631145792 model_lib_v2.py:708] {'loss/classification_loss': 0.0523538,
'loss/localization_loss': 0.009429479,
'loss/regularization_loss': 0.1346481,
'loss/total_loss': 0.19643137,
'learning_rate': 0.079357104}
INFO:tensorflow:Step 3900 per-step time 0.363s
10619 07:55:15.749054 139773631145792 model_lib_v2.py:705] Step 3900 per-step time 0.363s
INFO:tensorflow:{'loss/classification_loss': 0.051991254,
'loss/localization_loss': 0.01542333,
'loss/regularization_loss': 0.13386609,
'loss/total_loss': 0.20128067,
'learning_rate': 0.07931808}
10619 07:55:15.749407 139773631145792 model_lib_v2.py:708] {'loss/classification_loss': 0.051991254,
'loss/localization_loss': 0.01542333,
'loss/regularization_loss': 0.13386609,
'loss/total_loss': 0.20128067,
'learning_rate': 0.07931808}
INFO:tensorflow:Step 4000 per-step time 0.360s
10619 07:56:27.733394 139773631145792 model_lib_v2.py:705] Step 4000 per-step time 0.360s
INFO:tensorflow:{'loss/classification_loss': 0.042165227,
'loss/localization_loss': 0.011564374,
'loss/regularization_loss': 0.133809799,
'loss/total_loss': 0.1868276,
'learning_rate': 0.07926236}
10619 07:56:27.733752 139773631145792 model_lib_v2.py:708] {'loss/classification_loss': 0.042165227,
'loss/localization_loss': 0.011564374,
'loss/regularization_loss': 0.133809799,
'loss/total_loss': 0.1868276,
'learning_rate': 0.07926236}
```

Gambar 3. Hasil *Training* dalam 4000 iterasi

Proses pelatihan model membutuhkan waktu dan sumber daya komputasi yang cukup besar. Pastikan sistem memiliki keandalan dan performa yang cukup untuk menjalankan pelatihan secara efisien. Selain itu, perhatikan juga pengaturan *batch size* dan *learning rate* agar tidak terjadi divergensi atau konvergensi yang terlalu lambat. Pada pelatihan dengan 4000 iterasi, ditemukan bahwa pelatihan sudah cukup baik dan dapat dilanjutkan ke tahap evaluasi model.

2.2. Metrik Evaluasi yang digunakan

Dalam penelitian ini menggunakan beberapa metrik evaluasi untuk mengevaluasi kinerja model dalam identifikasi jenis sampah berdasarkan citra. Metrik pengujian yang digunakan adalah *Mean Average Precision (mAP)*, *Average Recall (AR)* dan *Intersect over Union (IoU)*.

Metrik *mAP* digunakan untuk mengukur akurasi keseluruhan sistem dalam mengidentifikasi jenis sampah dengan benar. Nilai *mAP* dihitung dengan mengambil rata-rata presisi pada berbagai level *recall*. Metrik ini memberikan gambaran tentang seberapa baik model dapat mengenali dan membedakan jenis sampah yang berbeda. Penghitungan *mAP* dilakukan dengan menghitung rata-rata dari *Average Precision (AP)*. Berikut adalah rumus *mAP* :

$$mAP = \frac{1}{n} \sum_{k=1}^{k=n} AP_k$$

$AP_k = \text{the AP of class } k$

$n = \text{the number of classes}$

Metrik AR mengukur tingkat pemulihan jenis sampah oleh sistem. AR dihitung dengan mengambil rata-rata $recall$ pada semua kelas sampah. Metrik ini memberikan gambaran tentang seberapa baik model mampu mengenali dan memulihkan jenis sampah yang ada dalam *dataset*. Berikut adalah rumus AR :

$$AR = \frac{1}{n} \sum_{k=1}^{k=n} Recall_k$$

Perhitungan *Precision* dilakukan dengan jumlah *True Positive* dibagi dengan *True Positive* ditambah *False Positive*. Perhitungan *Recall* dilakukan dengan jumlah *True Positive* dibagi dengan *True Positive* ditambah *False Negative*. Berikut adalah rumus *Precision* dan rumus *Recall*:

$$Precision = \frac{TP}{TP + FP} \quad Recall = \frac{TP}{TP + FN}$$

True Positive (TP) adalah kondisi dimana model dapat mendeteksi objek secara benar dan tepat. *False Positive (FP)* adalah kondisi ketika model dapat mendeteksi objek namun mendeteksi objek yang salah. *False Negative (FN)* adalah kondisi ketika model tidak dapat mendeteksi objek

3. HASIL

3.1. Sumber Data dan Pengumpulan

Sumber data yang digunakan dalam penelitian ini adalah dataset yang dikumpulkan secara langsung. Dataset ini mencakup sejumlah citra atau gambar yang mewakili berbagai jenis sampah yang ingin diidentifikasi oleh sistem. Proses pengumpulan data dilakukan dengan langkah-langkah berikut:

1. Identifikasi Jenis Sampah:
 - a. Tentukan jenis sampah yang akan diidentifikasi oleh sistem.
 - b. Jenis sampah yang digunakan pada penelitian ini adalah kertas, plastik, logam, kaca dan karton yang sesuai dengan tujuan penelitian.
 - c. Kertas : Kategori ini mencakup sampah yang terbuat dari kertas, seperti kertas koran, kertas bekas, atau kertas kemasan. Citra sampah kertas sering memiliki tekstur dan pola serat yang khas yang terlihat pada permukaan kertas. Sampah kertas mungkin memiliki bentuk yang bervariasi, seperti terlipat atau bertumpuk.
 - d. Plastik : Kategori ini mencakup sampah yang terbuat dari bahan plastik, seperti botol plastik, kantong plastik, atau wadah plastik lainnya. Citra-citra sampah plastik memiliki variasi warna, tekstur, dan bentuk, tergantung pada jenis plastiknya. Sampah plastik dapat berada dalam berbagai posisi dan bentuk, seperti tergeletak atau berbentuk wadah.
 - e. Logam : Kategori ini mencakup sampah yang terbuat dari bahan logam, seperti kaleng, tutup botol, atau barang-barang logam lainnya. Citra sampah logam ditandai dengan warna metalik atau cenderung memiliki permukaan yang

mengkilap. Sampah logam mungkin berada dalam posisi berbagai sudut dan orientasi.

- f. Kaca : Kategori ini mencakup sampah yang terbuat dari bahan kaca, seperti botol, kaca pecah, atau barang-barang kaca lainnya. Citra-citra kaca biasanya memiliki transparansi yang khas, dan terdapat pantulan cahaya yang memantul di permukaannya, terutama jika kaca memiliki permukaan yang licin dan berkilau. Sampah kaca dapat memiliki berbagai bentuk dan ukuran.
 - g. Karton : Kategori ini mencakup sampah yang terbuat dari karton, seperti kotak kemasan, kardus, atau wadah dari bahan karton. Citra sampah karton umumnya ditandai dengan permukaan berpola atau tekstur khas yang berasal dari serat-serat karton yang tampak jelas. Sampah karton mungkin berada dalam berbagai posisi, seperti berbaring, berdiri, atau bertumpuk.
2. Pengumpulan Sampel:
 - a. Melakukan pengumpulan sampel dari jenis sampah yang telah ditentukan.
 - b. Sampel diperoleh dari berbagai sumber, seperti rumah tangga, tempat pembuangan sampah, atau fasilitas daur ulang.
 3. Pengambilan Gambar:
 - a. Pengambilan gambar atau foto dari setiap sampel sampah yang dikumpulkan menggunakan kamera ponsel. Pada penelitian ini pengambilan gambar menggunakan ponsel Vivo seri 1901 dengan spesifikasi kamera 13MP+8MP+2MP.
 - b. Pengambilan gambar dilakukan dengan resolusi 720x720 pixel, dilakukan di dalam ruangan dengan pencahayaan yang memadai, dilakukan dengan latar belakang putih dan sudut pengambilan yang representatif. Dapat dilihat pada Gambar 4.



Gambar 4. Pengambilan Gambar

4. Jumlah dan Variasi Data:
 - a. Jumlah data pada penelitian ini adalah 180 data, dengan jumlah data per kategori adalah 36.
 - b. Variasi data per kategori adalah 6 variasi dengan sudut pengambilan yang beragam.
5. Kategorisasi Data:
 - a. Kategori data pada penelitian ini menggunakan penamaan dalam bahasa Inggris yaitu *cardboard*, *glass*, *metal*, *paper* dan *plastic*.

Setiap gambar diberi label dengan jenis sampah yang sesuai.

Setelah *dataset* sampah telah dianotasi dan dikategorikan, langkah selanjutnya adalah membagi data menjadi dua *subset*, yaitu data *training* dan data validasi. Pembagian ini berguna untuk melatih model *deep learning* dengan data *training* dan menguji performa model dengan data validasi. Berikut adalah langkah-langkah yang dilakukan:

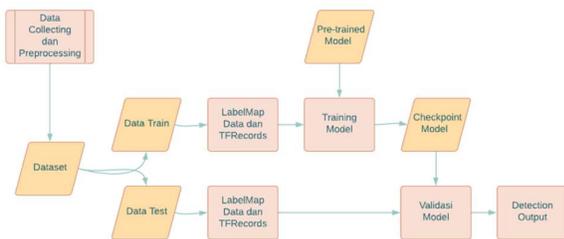
1. Penentuan Proporsi Pembagian:
 - a. Proporsi pembagian yang dilakukan adalah 85% data untuk *training* dan 15% data untuk validasi. Proporsi dilakukan pada setiap kelas agar jumlah data terdistribusi secara merata.
 - b. Proporsi ini disesuaikan tergantung pada jumlah data yang tersedia dan kebutuhan penelitian.
2. Pembagian Data secara Manual:
 - a. Melakukan pembagian data secara manual untuk menghindari bias dalam *subset training* dan validasi. Pengambilan data validasi dilakukan dengan mempertimbangkan data secara representatif.
 - b. Jumlah data pada data *training* adalah 150 data. Dan jumlah data pada data validasi adalah 30 data.
 - c. Jumlah data pada tiap kategori dapat dilihat pada tabel 1.

Tabel 1. Jumlah data pada tiap kategori

Data gambar	Jumlah data latih	Jumlah data uji	Jumlah data keseluruhan
<i>Cardboard</i>	30	6	36
<i>Glass</i>	30	6	36
<i>Metal</i>	30	6	36
<i>Paper</i>	30	6	36
<i>Plastic</i>	30	6	36
Total	150	30	180

3.2. Arsitektur Sistem

Diagram arsitektur sistem dapat dilihat pada Gambar 5.



Gambar 5. Diagram Arsitektur Sistem

Arsitektur sistem ini memungkinkan implementasi teknologi *deep learning* dan *TensorFlow Object Detection API* dalam mengidentifikasi jenis sampah berdasarkan citra. Dengan pendekatan *transfer learning*, diharapkan model dapat mencapai kinerja yang baik dalam mengklasifikasikan jenis sampah.

3.3. Hasil Evaluasi Model pada Data Validasi

Proses pengujian dilakukan menggunakan data uji yang telah disediakan sebanyak enam data per kelas dengan total data pengujian sebanyak 30 data. Pengujian dilakukan dengan *IoU threshold* sebesar 0.8. Hasil Pengujian dapat dilihat pada tabel 2.

Tabel 2. Hasil Uji Klasifikasi dan Deteksi *Cardboard*

No	Input	Klasifikasi	Status Klasifikasi	Deteksi (IoU)	Status Deteksi	Status Prediksi
1	<i>Cardboard-1</i>	-	Salah	-	Tidak Terdeteksi	FN
2	<i>Cardboard-2</i>	<i>Paper</i>	Salah	0.97	Terdeteksi	FP
3	<i>Cardboard-3</i>	<i>Cardboard</i>	Benar	0.99	Terdeteksi	TP
4	<i>Cardboard-4</i>	<i>Cardboard</i>	Benar	0.99	Terdeteksi	TP
5	<i>Cardboard-5</i>	<i>Cardboard</i>	Benar	0.98	Terdeteksi	TP
6	<i>Cardboard-6</i>	<i>Paper</i>	Salah	0.94	Terdeteksi	FP

Pada Tabel 2 dapat dilihat bahwa dari enam data *cardboard* yang diuji, tiga data dengan klasifikasi benar, dua data dengan klasifikasi salah dan satu data tidak terklasifikasi. Dari enam data uji *cardboard*, dapat dilihat lima data terdeteksi dengan tingkat deteksi yang tinggi dan satu data tidak terdeteksi. Dalam hal ini : *Precision* untuk kelas *cardboard* adalah :

$$Precision = \frac{TP}{TP + FP} = \frac{3}{3 + 2} = 0.6$$

Recall untuk kelas *cardboard* adalah :

$$Recall = \frac{TP}{TP + FN} = \frac{3}{3 + 1} = 0.75$$

Average Precision untuk kelas *cardboard* adalah :

$$AP = Precision/Recall = 0.6/0.75 = 0.8$$

Tabel 3. Hasil Uji Klasifikasi dan Deteksi *Glass*

No	Input	Klasifikasi	Status Klasifikasi	Deteksi (IoU)	Status Deteksi	Status Prediksi
1	<i>Glass-1</i>	<i>Glass</i>	Benar	0.99	Terdeteksi	TP
2	<i>Glass-2</i>	<i>Plastic</i>	Salah	0.96	Terdeteksi	FP
3	<i>Glass-3</i>	<i>Glass</i>	Benar	1	Terdeteksi	TP
4	<i>Glass-4</i>	<i>Plastic</i>	Salah	0.94	Terdeteksi	FP
5	<i>Glass-5</i>	<i>Glass</i>	Benar	0.88	Terdeteksi	TP
6	<i>Glass-6</i>	<i>Glass</i>	Benar	0.85	Terdeteksi	TP

Pada Tabel 3 dapat dilihat pada data uji, empat data dengan klasifikasi benar dan dua data dengan klasifikasi salah. Dari enam data uji *glass*, dapat dilihat enam data terdeteksi dengan tingkat deteksi yang tinggi. Dalam hal ini :

Precision untuk kelas *glass* adalah :

$$Precision = \frac{TP}{TP + FP} = \frac{4}{4 + 2} = 0.66$$

Recall untuk kelas *glass* adalah :

$$Recall = \frac{TP}{TP + FN} = \frac{4}{4 + 0} = 1$$

Average Precision untuk kelas *glass* adalah :

$$AP = Precision/Recall = 0.66/1 = 0.66$$

Setelah nilai *AP* pada setiap kelas dihitung untuk setiap jenis kategorinya. Tahapan selanjutnya adalah melakukan penghitungan nilai *mAP* seluruh kelas dengan menjumlahkan nilai *AP* setiap kelas dibagi banyak kelas. Dalam hal ini:

Perhitungan *mAP* seluruh kelas:

$$mAP = \frac{1}{5} \sum AP = \frac{1}{5} (0.8 + 0.66 + 0.83 + 1 + 1) = \frac{1}{5} (4.29) = 0.858$$

Jumlah *mAP* pada seluruh kelas data uji adalah sebesar 0.858. Ini menandakan bahwa akurasi model sudah baik. Hasil *mAP* pengujian juga sudah mendekati nilai evaluasi model pada proses evaluasi model. Selanjutnya, melakukan penghitungan nilai *AR*

seluruh kelas dengan menjumlahkan nilai *Recall* setiap kelas dibagi banyak kelas. Dalam hal ini:

Perhitungan *AR* seluruh kelas:

$$AR = \frac{1}{5} \sum Recall = \frac{1}{5} (0.75 + 1 + 0.83 + 1 + 1) \\ = \frac{1}{5} (4.58) = 0.916$$

Jumlah *AR* pada seluruh kelas data uji adalah sebesar 0.916. Ini menandakan bahwa tingkat pemulihan model sudah sangat baik. Dengan menggunakan *IoU threshold* sebesar 0.8, model dapat memberikan *score* deteksi yang diatas nilai *threshold* tersebut. Hal ini menandakan bahwa model dapat menentukan posisi objek dengan sangat baik dan akurat.

3.4. Hasil Evaluasi Model pada Dataset yang Belum Dilihat Sebelumnya

Proses pengujian yang dilakukan menggunakan data yang belum pernah dilihat sebelumnya. Pengujian dilakukan dengan *IoU threshold* sebesar 0.84. Hasil Pengujian dapat dilihat pada tabel 4.

Dengan 20 data yang diuji, 15 data dengan klasifikasi benar, empat data dengan klasifikasi salah dan satu data tidak terklasifikasi. Dari 20 data uji, dapat dilihat 19 data terdeteksi dengan tingkat deteksi yang tinggi dan satu data tidak terdeteksi. Dalam hal ini :

Precision untuk kelas *paper* adalah :

$$Precision = \frac{TP}{TP + FP} = \frac{15}{15 + 4} = 0.789$$

Recall untuk kelas *paper* adalah :

$$Recall = \frac{TP}{TP + FN} = \frac{15}{15 + 1} = 0.937$$

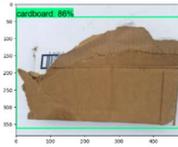
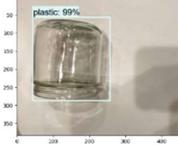
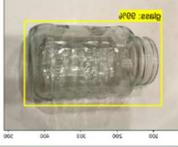
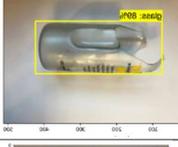
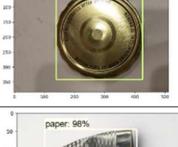
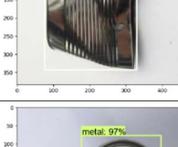
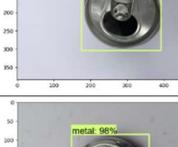
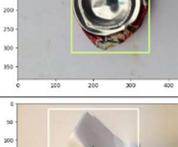
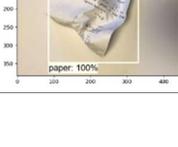
Average Precision untuk kelas *paper* adalah :

$$AP = Precision/Recall = 0.789/0.937 = 0.842$$

Jumlah *AP* pada seluruh data yang uji adalah sebesar 0.842. Tidak jauh berbeda dengan pengujian pada data validasi. Ini menandakan bahwa kemampuan model dalam mendeteksi jenis sampah sudah baik dan tidak terjadi *overfitting*. Namun, terjadi penurunan akurasi, hal ini wajar dikarenakan model mencoba mendeteksi data yang benar-benar belum pernah dilihatnya pada data latih.

Pada pengujian dengan data ini, dapat dilihat bahwa akurasi kelas kertas dan plastik memiliki tingkat akurasi yang tinggi karena memiliki karakteristik yang unik sehingga model bisa mengenali dengan tepat. Namun, pada kelas kaca memiliki akurasi yang rendah dikarenakan karakteristik gambar kaca yang mirip dengan plastik sehingga model kurang bisa mengenali kelas kaca dengan baik. Berbeda dengan kelas karton dan logam, walaupun memiliki karakter yang cukup unik model masih salah dalam mengenali beberapa data gambar. Kemungkinan hal ini terjadi dikarenakan kurangnya pengambilan data yang representatif baik dalam sudut pengambilan maupun pencahayaan.

Tabel 4. Hasil Evaluasi Model pada Dataset yang Belum Dilihat Sebelumnya

No	Input	Klasifikasi	Status Klasifikasi	Deteksi (IoU)	Status Deteksi
1		Cardboard	Benar	0.86	Terdeteksi
2		-	Salah	-	Tidak Terdeteksi
3		Plastic	Salah	0.99	Terdeteksi
4		Glass	Benar	0.99	Terdeteksi
5		Glass	Benar	0.89	Terdeteksi
6		Metal	Benar	0.96	Terdeteksi
7		Paper	Salah	0.98	Terdeteksi
8		Metal	Benar	0.97	Terdeteksi
9		Metal	Benar	0.96	Terdeteksi
10		Paper	Benar	1	Terdeteksi

4. PEMBAHASAN

Dalam sub-bab ini, akan dilakukan evaluasi terhadap sistem secara keseluruhan, termasuk keberhasilan identifikasi jenis sampah oleh sistem dan keberhasilan integrasi model ke dalam sistem. Berikut adalah evaluasi sistem secara keseluruhan:

4.1. *Keberhasilan Identifikasi Jenis Sampah oleh Sistem*

Sistem yang dikembangkan telah berhasil dalam mengidentifikasi jenis sampah berdasarkan citra dengan tingkat akurasi yang tinggi. Melalui penggunaan teknologi *deep learning* dan *transfer learning*, sistem mampu mempelajari pola-pola yang ada pada data pelatihan dan menggeneralisasikannya pada data baru. Hal ini memungkinkan sistem untuk mengenali dan membedakan berbagai jenis sampah dengan akurasi yang baik. Keberhasilan identifikasi sampah ini memiliki implikasi yang signifikan dalam konteks pengelolaan sampah, di mana sistem dapat membantu dalam proses pengelompokan dan pengolahan sampah dengan lebih efisien.

4.2. *Keberhasilan Integrasi Model ke dalam Sistem*

Model yang telah dilatih dengan menggunakan *TensorFlow Object Detection API* berhasil diintegrasikan ke dalam sistem dengan sukses. Integrasi ini memungkinkan sistem untuk menggunakan model dalam waktu nyata untuk mengidentifikasi jenis sampah pada citra yang diunggah oleh pengguna. Sistem mampu mengambil *input* citra, menerapkan model identifikasi, dan menghasilkan *output* yang menampilkan jenis sampah yang teridentifikasi. Keberhasilan integrasi model ini menunjukkan kesesuaian antara model *deep learning* yang dikembangkan dan sistem yang dirancang.

4.3. *Diskusi Keterbatasan dan Tantangan yang Ditemui*

Dalam implementasi sistem identifikasi jenis sampah berbasis *deep learning*, terdapat beberapa keterbatasan dan tantangan yang dihadapi. Pemahaman akan keterbatasan ini penting untuk memahami batasan sistem dan mempertimbangkan langkah-langkah perbaikan atau pengembangan di masa depan. Salah satu keterbatasan yang dihadapi adalah keterbatasan dalam kualitas dan jumlah data pelatihan. Meskipun telah dilakukan upaya yang signifikan untuk mengumpulkan dan menganotasi data sampah, masih mungkin terdapat variasi yang terbatas dalam jenis sampah yang direpresentasikan oleh data pelatihan. Keterbatasan ini dapat mempengaruhi kinerja model identifikasi pada jenis sampah yang jarang atau tidak ada dalam data pelatihan. Oleh karena itu, penambahan data pelatihan yang lebih beragam dapat menjadi langkah perbaikan di masa depan.

Selain itu, keterbatasan perangkat keras dan kecepatan komputasi juga dapat menjadi tantangan dalam implementasi sistem. Proses pelatihan dan inferensi model *deep learning* membutuhkan daya komputasi yang cukup tinggi, terutama jika dilakukan pada *dataset* yang besar. Keterbatasan ini dapat mempengaruhi waktu yang dibutuhkan untuk melatih model dan kecepatan respons sistem saat melakukan identifikasi sampah. Untuk mengatasi

tantangan ini, penggunaan perangkat keras yang lebih kuat atau pengoptimalan algoritma dapat menjadi langkah perbaikan di masa depan.

Selain keterbatasan tersebut, keterbatasan dalam lingkungan operasional juga dapat mempengaruhi kinerja sistem. Variasi pencahayaan, posisi objek, atau kualitas citra dapat mempengaruhi kemampuan sistem untuk mengidentifikasi jenis sampah dengan akurat. Selain itu, adanya elemen gangguan atau *noise* dalam citra juga dapat mempengaruhi kinerja sistem. Mengatasi keterbatasan ini dapat melibatkan pengembangan metode pemrosesan citra yang lebih adaptif atau penggunaan teknik pemrosesan sinyal yang lebih canggih.

Tantangan lain yang dihadapi adalah pemeliharaan dan pembaruan sistem. Sistem yang diimplementasikan perlu diperbarui secara berkala dengan pembaruan model identifikasi yang lebih baik atau konfigurasi yang dioptimalkan. Selain itu, perawatan perangkat keras dan perangkat lunak juga menjadi penting untuk menjaga keberlanjutan dan keandalan sistem. Dalam menghadapi tantangan ini, perencanaan yang baik untuk pemeliharaan dan pembaruan sistem perlu diperhatikan. Meskipun ada keterbatasan dan tantangan yang dihadapi dalam implementasi sistem, langkah-langkah perbaikan dan pengembangan yang tepat dapat diambil untuk mengatasi kendala tersebut. Dalam mengembangkan sistem identifikasi sampah berbasis *deep learning*, penting untuk terus melakukan penelitian dan eksperimen guna meningkatkan kualitas dan kinerja sistem secara keseluruhan.

5. KESIMPULAN

Model *deep learning* yang dibangun menggunakan *TensorFlow Object Detection API* dan *transfer learning* telah terbukti mampu mengidentifikasi jenis sampah pada sampah umum di Indonesia dengan akurasi yang tinggi. Model ini berhasil mengklasifikasikan sampah menjadi lima kelas utama: kaca, karton, kertas, logam, dan plastik.

Model ini telah menunjukkan kinerja yang sangat baik dalam mengenali dan mengklasifikasikan sampah. Dalam pengujian menggunakan data uji, model mencapai nilai Mean Average Precision (mAP) sebesar 0.858, Average Recall (AR) sebesar 0.91, dan nilai Intersect over Union (IoU) sebesar 0.841. Hal ini mengindikasikan kemampuan model dalam mengenali dan membedakan jenis sampah dengan tingkat presisi yang tinggi.

Penerapan *TensorFlow Object Detection API* dan *transfer learning* bukan hanya memungkinkan pengembangan model yang efisien, tetapi juga mempercepat proses pelatihan model. Ini adalah langkah penting dalam meningkatkan kinerja model dalam identifikasi jenis sampah.

Penggunaan metode pembagian data sebesar 85% untuk data *training* dan 15% untuk data uji terbukti memberikan hasil yang baik. Dengan jumlah data keseluruhan sebanyak 180 data, model memiliki basis data yang konsisten dan dapat diandalkan. Data *training* terdiri dari 150 data, sementara data uji terdiri dari 30 data.

Model berhasil dilatih dalam 4000 iterasi dengan waktu pelatihan yang relatif singkat, yakni sekitar 13 menit 40 detik. Ini menunjukkan efisiensi dalam proses pelatihan model.

Visualisasi grafik menggunakan *TensorBoard* memberikan pemantauan yang baik terhadap pelatihan model secara *real-time*. Model juga berhasil diekspor ke format *TFLite*, sehingga dapat diimplementasikan pada perangkat dengan sumber daya terbatas. Implementasi model menggunakan data statis dan *OpenCV* juga memungkinkan penggunaan model untuk mengidentifikasi jenis sampah secara *real-time* melalui *input* kamera.

DAFTAR PUSTAKA

- [1] F. Kaza S. ., Yao, L. ., Bhada-Tata, P. ., & Van Woerden, *What a waste 2.0: a global snapshot of solid waste management to 2050*. World Bank Publications, 2018.
- [2] V. Sudha S. ., Vidhyalakshmi, M. ., Pavithra, K. ., Sangeetha, K. . & Swaathi, "An automatic classification method for environment: Friendly waste segregation using deep learning," *IEEE*, 2016, pp. 65–70.
- [3] H. Wang, "Garbage recognition and classification system based on convolutional neural network vgg16," *IEEE*, 2020, pp. 252–255.
- [4] S. de Mello Soares C. T. ., Ek, M. ., Östmark, E. ., Gällstedt, M. ., & Karlsson, "Recycling of multi-material multilayer plastic packaging: Current trends and future scenarios," *Resources, conservation and recycling*, p. 105905, 2022.
- [5] Z. Adedeji O. . & Wang, "Intelligent waste classification system using deep learning convolutional neural network," *Procedia Manufacturing*, pp. 607–612, 2019.
- [6] S. White G. ., Cabrera, C. ., Palade, A. ., Li, F. . & Clarke, "WasteNet: Waste classification at the edge for smart bins," *arXiv*, 2020.
- [7] J. Song F. ., Zhang, Y. ., & Zhang, "Optimization of CNN-based garbage classification model," 2020, pp. 1–5.
- [8] S. Longo E. ., Sahin, F. A. ., Redondi, A. E. ., Bolzan, P. ., Bianchini, M. . & Maffei, "Take the trash out... to the edge. Creating a Smart Waste Bin based on 5G Multi-access Edge Computing," 2021, pp. 55–60.
- [9] P. Thanawala D. ., Sarin, A. . & Verma, "An approach to waste segregation and management using convolutional neural networks," Springer Singapore, 2020, pp. 139–150.
- [10] M. T. Sheng T. J. ., Islam, M. S. ., Misran, N. ., Baharuddin, M. H. ., Arshad, H. ., Islam, M. R. ., . . . & Islam, "An internet of things based smart waste management system using LoRa and tensorflow deep learning model," *IEEE Access*, pp. 148793–148811, 2020.
- [11] J. Bobulski and M. Kubanek, "Deep learning for plastic waste classification system," *Applied Computational Intelligence and Soft Computing*, vol. 2021, pp. 1–7, 2021.
- [12] M. Malik *et al.*, "Waste classification for sustainable development using image recognition with deep learning neural network models," *Sustainability*, vol. 14, no. 12, p. 7222, 2022.
- [13] N. Ramsurrun, G. Suddul, S. Armoogum, and R. Foogooa, "Recyclable waste classification using computer vision and deep learning," presented at the 2021 zooming innovation in consumer technologies conference (ZINC), IEEE, 2021, pp. 11–15.
- [14] P. H. Sunanto O. D. S. ., & Utomo, "Implementasi Deep Learning Dengan Convolutional Neural Network Untuk Klasifikasi Gambar Sampah Organik Dan Anorganik," *UNEJ e-Proceeding*, pp. 373–382, 2022.
- [15] H. Kumar and A. Martin, "Artificial Emotional Intelligence: Conventional and deep learning approach," *Expert Systems with Applications*, vol. 212, p. 118651, Feb. 2023, doi: 10.1016/j.eswa.2022.118651.
- [16] S. Kumar *et al.*, "Deep learning framework for recognition of cattle using muzzle point image pattern," *Measurement*, vol. 116, pp. 1–17, 2018.
- [17] M. Arsal, B. A. Wardijono, and D. Anggraini, "Face Recognition Untuk Akses Pegawai Bank Menggunakan Deep Learning Dengan Metode CNN," *J. Nas. Teknol. dan Sist. Inf*, vol. 6, no. 1, pp. 55–63, 2020.
- [18] D. N. Melinte D. O. ., Travediu, A. M. . & Dumitriu, "Deep convolutional neural networks object detector for real-time waste identification," *Applied Sciences*, p. 7301, 2020.