



PENERAPAN MODEL ARSITEKTUR VGG16 UNTUK KLASIFIKASI JENIS SAMPAH

Ety Sutanty¹⁾, Maukar^{1)*}, Dina Kusuma Astuti¹⁾, Handayani¹⁾

¹⁾Universitas Gunadarma, Depok, Indonesia

Email: maukar@staff.gunadarma.ac.id

Abstrak

Data dari Kementerian Lingkungan Hidup dan Kehutanan melalui Sistem Informasi Pengelolaan Sampah Nasional menunjukkan peningkatan jumlah tumpukan sampah hampir 30 Ton pada 2021 dengan sumber sampah terbanyak berasal dari produksi sampah rumah tangga dan kegiatan industri. Cara menyelesaikan masalah pengolahan sampah adalah melalui kegiatan pengolahan sampah yang lebih efektif, salah satunya dengan melakukan klasifikasi jenis sampah menggunakan *Convolutional Neural Network (CNN)*. Pada penelitian ini, dilakukan klasifikasi jenis sampah berdasarkan citra sampah dengan metode CNN menggunakan arsitektur VGG-16 menggunakan dataset *Garbage Classification* dari situs Kaggle. Arsitektur VGG-16 pada penelitian ini menggunakan 16 lapisan *layer* yang terdiri atas 13 lapisan *layer* konvolusi dan 3 lapisan *fully-connected*. Pelatihan model menggunakan *library tensorflow* dengan rasio split data 7:1.5:1.5 untuk data latih, data test dan data validasi. Model CNN arsitektur VGG-16 sebagai metode *transfer learning* dimodifikasi dengan penambahan beberapa lapisan diantaranya *pooling layer* dengan *GlobalAveragePooling2D*, *dense layer* dengan aktivasi ReLU, dan *dense layer* dengan aktivasi *softmax* sebagai lapisan *full connected layer*. Hasil dari pelatihan mendapatkan nilai *accuracy* sebesar 82.89% dan nilai *val_accuracy* sebesar 84.62%. Hasil penelitian ini diharapkan dapat membantu pemerintah dalam melakukan pengolahan sampah dalam bentuk yang lebih stabil dan tidak mencemari lingkungan serta mengurangi jumlah sampah yang harus ditimbun.

Kata kunci: arsitektur; sampah; *transfer learning*; *tensorflow*; VGG16.

IMPLEMENTATION OF VGG16 ARCHITECTURE MODEL FOR CLASSIFICATION OF GARBAGE

Abstract

Data from the Ministry of Environment and Forestry through the National Waste Management Information System shows an increase in the amount of waste piled up by almost 30 tonnes in 2021 with the largest sources of waste coming from household waste production and industrial activities. The way to solve the waste processing problem is through more effective waste processing activities, one of which is by classifying waste types using a *Convolutional Neural Network (CNN)*. In this research, classification of waste types was carried out based on waste images using the CNN method using the VGG-16 architecture using the *Garbage Classification* dataset from the Kaggle site. The VGG-16 architecture in this research uses 16 layers consisting of 13 convolution layers and 3 fully-connected layers. Model training uses the *TensorFlow* library with a data split ratio of 7:1.5:1.5 for training data, test data and validation data. The VGG-16 architectural CNN model as a *transfer learning* method was modified by adding several layers including a *pooling layer* with *GlobalAveragePooling2D*, a *dense layer* with ReLU activation, and a *dense layer* with *softmax* activation as a *full connected layer*. The results of the training obtained an *accuracy* value of 82.89% and a *val_accuracy* value of 84.62%. It is hoped that the results of this research can help the government process waste in a form that is more stable and does not pollute the environment and reduce the amount of waste that must be landfilled.

Keywords: architecture; garbage; *transfer learning*; *tensorflow*; VGG16.

Submitted: 5 September 2023	Reviewed: 6 September 2023	Accepted: 25 September 2023	Published: 25 September 2023
--------------------------------	-------------------------------	--------------------------------	---------------------------------

PENDAHULUAN

Sampah merupakan hasil dari kegiatan non biologis manusia dan didefinisikan sebagai material sisa yang tidak disenangi dan tidak diinginkan oleh manusia (Satispi & Aziz Samudra, 2022) apabila masa penggunaannya telah berakhir. Sedangkan timbulan sampah adalah banyaknya sampah yang timbul dari masyarakat dalam satuan volume maupun berat per kapita perhari (Aprilia, 2021), atau perluas bangunan. Berdasarkan sifatnya jenis sampah (Amasuomo & Baird, 2016) dibagi ke dalam tiga kelompok, yaitu sampah organik, sampah anorganik dan sampah B3 (Bahan Berbahaya dan Beracun). Sampah organik memiliki karakteristik berasal dari makhluk hidup dan merupakan sampah basah (Fang et al., 2023). Contoh dari sampah organik adalah dedaunan, sampah dapur, buah-buahan, dan kotoran hewan. Sampah organik memiliki istilah lain yakni sampah *degradable* (Parkar et al., 2021) karena kemudahannya untuk terurai secara alami tanpa melibatkan campur tangan manusia. Berkebalikan dengan sampah organik, sampah anorganik merupakan sampah kering dan sangat sulit untuk diuraikan (*nondegradable*) seperti : plastik, karet, kaca, dan logam (Rahmayanti, et al, 2018), sedangkan sampah B3 yaitu sampah yang mengandung bahan berbahaya dan beracun sehingga secara langsung maupun tidak langsung dapat merusak lingkungan (Ichtiakhiri & Sudarmaji, 2015), mengganggu kesehatan, dan mengancam kelangsungan hidup manusia serta organisme lainnya, seperti : baterai, masker medis, deterjen pakaian dan popok bayi. Penimbunan sampah akan berdampak dengan permasalahan lingkungan dan kesehatan, sehingga perlu dilakukan langkah penanganan yang serius salah satunya adalah dengan melakukan klasifikasi jenis sampah. Kegiatan pengolahan sampah yang lebih efektif, salah satunya yaitu dengan melakukan daur ulang sampah dimana untuk melakukan daur ulang sampah, perlu adanya pemilahan sampah sesuai dengan jenisnya. Selama ini tempat sampah yang tersedia hanya berupa tempat sampah dengan pemilahan yang dilakukan oleh masyarakat secara manual. Selain itu tidak semua masyarakat memahami perbedaan jenis-jenis sampah yang dibuang. Tempat sampah yang dipisah menjadi organik, anorganik dan B3 berguna untuk memudahkan proses mendaur ulang namun beberapa masyarakat memiliki kebiasaan dalam mencampur jenis-jenis sampah yang bahkan dapat berbahaya apabila digabungkan menjadi satu, sehingga klasifikasi menjadi hal penting sebagai upaya dalam melakukan pemilahan sampah. Cara untuk melakukan klasifikasi terhadap jenis sampah adalah dengan mengimplementasikan penggunaan *Convolutional Neural Network* (CNN).

Beberapa penelitian terdahulu terkait implementasi CNN dalam klasifikasi jenis sampah dilakukan peneliti terdahulu. Penelitian (Kohsasih et al., 2021) melakukan analisis perbandingan kinerja algoritma CNN dengan algoritma MLP dalam melakukan klasifikasi jenis sampah. Model yang dibangun (Kohsasih et al., 2021) menggunakan algoritma CNN mendapatkan performa yang lebih baik dimana hasil *precision*, *recall*, *f1-score*, dan *accuracy* sebesar 0.98, sedangkan model yang dibangun menggunakan algoritma MLP hanya mendapatkan *accuracy* sebesar 0.43, *precision* 0.41, *recall* 0.43 dan *f1-score* sebesar 0.39. Pada proses pelatihan dan pengujian model yang dibangun menggunakan algoritma CNN membutuhkan waktu yang lebih lama dibandingkan algoritma MLP. Selain itu model yang dibangun menggunakan algoritma CNN lebih efektif dalam melakukan klasifikasi sampah berdasarkan kelasnya. Penelitian (Stephen et al., 2019) mengimplementasikan penggunaan CNN dengan metode *transfer learning* untuk melakukan klasifikasi jenis sampah. *Transfer learning* dalam penelitian ini menggunakan *pre-trained* model dari ImageNet. Hasil penelitian menunjukkan bahwa model CNN yang paling baik adalah Resnet50. Nilai akurasi yang didapatkan dari train adalah 78% dan 90%, sedangkan nilai akurasi dari *validation* sebesar 74% dan 80%. Dalam penelitian ini untuk dapat menerapkan teknologi pemisahan sampah secara otomatis pada tempat sampah maka dapat menggunakan model Resnet50 apabila menggunakan *server* atau komputer dengan spesifikasi yang tinggi. Penelitian (Ibnul Rasidi et al., 2022)

melakukan pengelolaan sampah dengan mengidentifikasi jenis sampah dengan CNN. Hasil penelitian (Ibnul Rasidi et al., 2022) menunjukkan angka ketepatan sebesar 96% untuk sampah anorganik dan 62% untuk sampah organik. Penelitian (William Hutamaputra, Rifky Yunus Krisnabayu, Marrisaeka Mawarni, Novanto Yudistira, 2022) mengusulkan model VGG-16 dan ResNet34) untuk mengidentifikasi dan mengklasifikasikan sampah botol. Berdasarkan hasil pengujian menggunakan CNN, arsitektur VGG-16 memiliki akurasi sebesar 90% dan ResNet34 memiliki akurasi sebesar 50% pada klasifikasi botol plastik dan bukan botol plastik. Masing-masing arsitektur menggunakan 10 epoch, 32 *batch* dari sejumlah 1655 gambar.

Penelitian ini mengimplementasikan penggunaan CNN dengan arsitektur VGG16 dengan 16 lapisan *layer* yang terdiri atas 13 lapisan *layer* konvolusi dan 3 lapisan *fully-connected*. Dataset pada penelitian ini menggunakan citra sampah *Garbage Classification Kaggle* (Classification, 2018). Pembuatan model klasifikasi menggunakan dua *library* utama yaitu Keras sebagai *high-level neural network API* dan TensorFlow sebagai *backend engine*. Arsitektur VGG-16. digunakan sebagai metode transfer learning dimana model tersebut sudah dilatih terhadap suatu dataset untuk menyelesaikan masalah yang serupa dengan cara digunakan sebagai titik permulaan dengan memodifikasi atau mengubah parameternya sehingga dapat sesuai dengan dataset baru. Hasil penelitian ini diharapkan dapat mengklasifikasikan sampah berdasarkan jenisnya untuk memudahkan pembuangan dan pengolahan kembali, memisahkan pembuangan sampah organik dan non-organik sehingga menghindari terjadinya penumpukan sampah sebagai rangkaian kegiatan dalam pengelolaan sampah yang lebih baik.

METODE

Klasifikasi jenis sampah menggunakan algoritma *Convolutional Neural Network* (CNN) dengan arsitektur VGG16 dilakukan dengan beberapa tahapan proses. Dataset yang digunakan berisi citra jenis sampah yang bersumber dari dataset *Garbage Classification* (Classification, 2018). Setiap citra yang digunakan berformat .jpg dengan total citranya berjumlah 8134 data. Dataset tersebut terbagi menjadi 10 kelas berdasarkan jenis sampah seperti dapat dilihat pada Tabel 1.

Tabel 1. Kelas Jenis Sampah Dataset *Garbage Classification* (Classification, 2018)

No.	Kelas Sampah	Total Data
1	anorganic_brown-glass	607
2	anorganic_cardboard	891
3	anorganic_green-glass	629
4	anorganic_metal	769
5	anorganic_paper	1050
6	anorganic_plastic	865
7	anorganic_white-glass	775
8	b3_battery	945
9	b3_diaper_mask	620
10	organic	985

Terdapat beberapa tahap yang akan dilakukan untuk melakukan perancangan program klasifikasi jenis sampah dengan mengimplementasikan penggunaan CNN menggunakan

arsitektur VGG-16 pada citra sampah. Tahapan perancangan mencakup pembagian dataset, *pre-processing* data dan pembentukan model CNN dengan arsitektur VGG-16.

Pembagian dataset dilakukan setelah proses pengumpulan *dataset*, *dataset* perlu dibagi menjadi 3 bagian yaitu data *train*, data *validation*, dan data *test*. Data *train* digunakan untuk membentuk model klasifikasi (melatih) yang berisikan pengetahuan atau fitur dari citra sampah. Data *validation* digunakan untuk proses validasi model dan untuk melihat performa dari model CNN yang telah dibuat. Data *test* digunakan sebagai simulasi penggunaan model pada dunia nyata, data *test* tidak boleh dikenali sebelumnya oleh model yang telah dibuat. *Dataset* akan dibagi dengan perbandingan data *train* sebesar 70%, data *validation* sebesar 15%, dan data *test* sebesar 15% mengacu pada ujicoba yang menunjukkan rasio ini merupakan rasio split dengan akurasi terbaik seperti dapat dilihat pada Tabel 2.

Tabel 2. Split Rasio Dataset

No	Rasio Train:Test:Val	Accuracy	Error
1	8:1:1	73.56%	26.44%
2	7:1.5:1.5	82.89%	17.11
3	6:2:2	81.15%	18.85

Pembagian dataset dilakukan menggunakan *pseudocode*:

```
!pip install split-folders
splitfolders.ratio(base_ds, output='imgs', seed=123, ratio=(.7,.15,.15), group_prefix=None)
```

Parameter *install split-folders* melakukan instalasi module *split-folders* yang digunakan untuk melakukan pembagian *dataset* dengan *ratio*=(.7,.15,.15) yang menunjukkan bahwa pembagian dataset dengan perbandingan 70% untuk data *train*, 15% untuk data *validation*, dan 15% untuk data *test*. Tahapan *preprocessing* data dilakukan setelah *dataset* yang digunakan telah dibagi menjadi data *train*, data *validation*, dan data *test*. Pada tahap *pre-processing* data, setiap data akan dilakukan proses *rescale* dan augmentasi data untuk keperluan memperbanyak dan mengumpulkan dataset yang lebih bervariasi atau beragam. Proses augmentasi gambar ini diantaranya *shear_range*, *rotation_mode*, *fill_mode*, *zoom_range*, *width_shift_range*, *height_shift_range*, *horizontal_flip*, *vertical_flip*. Pada *pre-processing* data *train*, akan menerapkan *rescale* dan augmentasi data menggunakan *pseudocode*:

```
# Create Training Generator
train_datagen = ImageDataGenerator( rescale=1./255,
    shear_range=0.2, rotation_range=30,
    fill_mode='nearest', width_shift_range=0.2,
    height_shift_range=0.2, horizontal_flip=True,
    vertical_flip=True )
```

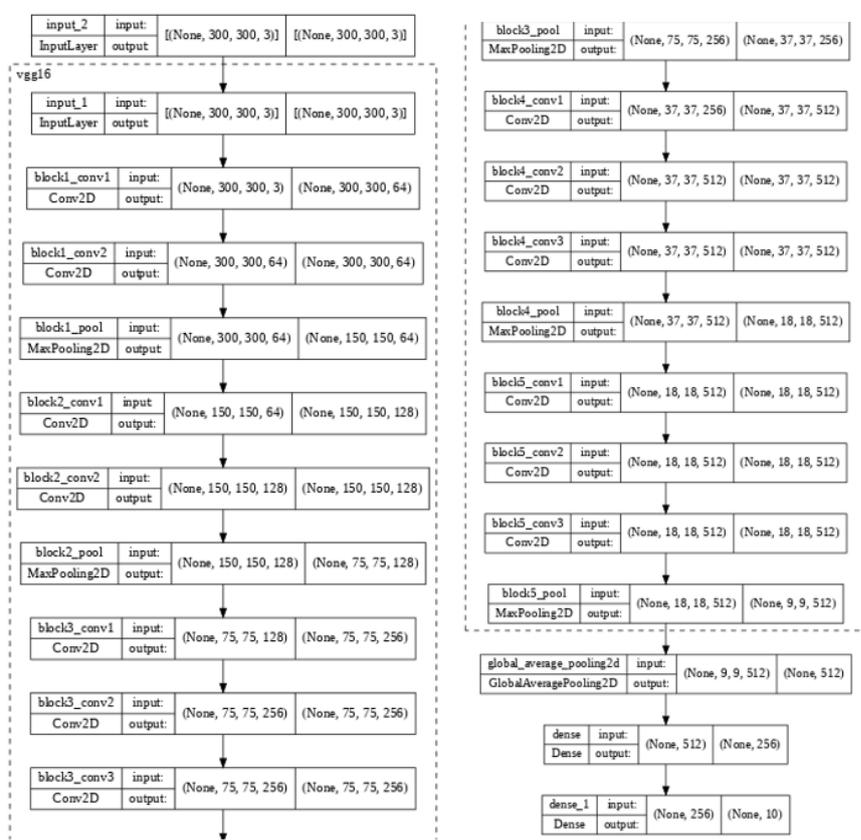
Variabel *train_datagen* variabel yang digunakan untuk menyimpan hasil *rescale* dan augmentasi data *train*, kemudian digunakan *rescale*=1./255 digunakan untuk mengubah ukuran skala nilai citra menjadi antara 0 dan 1 (Wulandari et al., 2023) sehingga data dari tiap citra akan seragam rentang nilainya. Parameter *shear_range*=0.2 digunakan untuk menerapkan transformasi geser secara acak dengan rentang nilai sampai 0.2 (Mittal et al., 2020), *rotation_range*=30 digunakan untuk menerapkan rotasi citra dengan skala 0-30 derajat secara acak. Penggunaan *fill_mode*='nearest' digunakan untuk pengisian area citra yang hilang atau kosong dengan nilai disekitarnya, *zoom_range*=0.2 digunakan untuk menerapkan transformasi

memperbesar gambar secara acak dengan rentang nilai sampai 0.2, *width_shift_range=0.2* digunakan untuk menerapkan pergeseran citra secara horizontal secara acak dengan rentang nilai sampai 0.2, *height_shift_range=0.2* digunakan untuk menerapkan pergeseran citra secara vertikal secara acak dengan rentang nilai sampai 0.2, *horizontal_flip=True* digunakan untuk membalikan citra secara horizontal dan *vertical_flip=True* digunakan untuk membalikan citra secara vertikal. Setelah melakukan proses *rescale* dan augmentasi data terhadap data *train*, langkah selanjutnya yaitu membuat *train generator* yang berisikan informasi mengenai *flow* data yang digunakan yaitu data *train* (Malik et al., 2022). Pembuatan *train generator* menggunakan *pseudocode*:

```

train_generator = train_datagen.flow_from_directory(
    'imgs/train', target_size = (img_height, img_width),
    batch_size = batch_size, subset = "training",
    class_mode='categorical', shuffle=True,
    color_mode='rgb')
    
```

Variabel *'imgs/train'* merupakan direktori yang mengarahkan kepada folder data *train* dengan *target_size* merupakan dimensi dari citra yang akan digunakan dalam proses pelatihan yaitu sebesar 300x300 piksel. Model klasifikasi citra sampah menggunakan *batch_size* merupakan banyaknya citra yang dimasukkan dalam setiap *steps training* yaitu sebesar 64 *batch* dengan *class_mode='categorical'* merupakan metode pemilihan klasifikasi yaitu klasifikasi secara kategorikal. Variabel *shuffle=True* data citra didalam folder tersebut akan diacak sehingga tidak sesuai urutan yang ada seperti urutan alfabetik dengan *color_mode='rgb'* yang merupakan tipe citra yang digunakan yaitu citra berwarna. Pembuatan model CNN dalam melakukan identifikasi dan klasifikasi fitur ciri pada citra sampah dapat dilihat pada Gambar 1.



Gambar 1. Model Arsitektur VGG16 untuk Klasifikasi Jenis Sampah

Penelitian ini menggunakan dua *library* utama yaitu Keras sebagai *high-level neural network* API dan TensorFlow sebagai *beckend engine*. Model dibuat menggunakan metode CNN dengan arsitektur VGG-16. Arsitektur VGG-16 digunakan sebagai metode *transfer learning* dimana model tersebut sudah dilatih terhadap suatu dataset untuk menyelesaikan masalah yang serupa dengan cara digunakan sebagai titik permulaan dengan memodifikasi atau mengubah parameternya sehingga dapat sesuai dengan *dataset* baru. Pembuatan model CNN dengan arsitektur VGG-16 untuk klasifikasi jenis sampah menggunakan *pseudocode*:

```

vgg16 = VGG16(weights="imagenet", include_top=False, input_shape=input_shape)
vgg16.trainable = False
inputs = tf.keras.Input(input_shape)
x = vgg16(inputs, training=False)
x = tf.keras.layers.GlobalAveragePooling2D()(x)
x = tf.keras.layers.Dense(256, activation='relu')(x)
x = tf.keras.layers.Dense(10, activation='softmax')(x)
model = tf.keras.Model(inputs, x)

```

Parameter *VGG16* merupakan arsitektur CNN yang digunakan untuk pembuatan model klasifikasi jenis sampah menggunakan *weights="imagenet"* yang merupakan parameter *weights* dari VGG16, dengan parameter *weights*-nya yaitu *imagenet*. Variabel *include_top=False* menandakan bahwa model tidak menggunakan *Full-Connected Layer* dari VGG16, sehingga jika melakukan *predict* maka yang akan terjadi adalah dataset akan mengalir pada *feature extraction layer* dari VGG16 (Poudel & Poudyal, 2022). Model klasifikasi penelitian ini menggunakan *input_shape* berukuran (300, 300, 3). Model arsitektur VGG16 yang telah dibuat kemudian dimodifikasi dengan penambahan beberapa lapisan, yaitu: *GlobalAveragePooling2D*, *Dense Layer*, menggunakan aktivasi *Rectified Linear Unit* (ReLU) dengan 256 parameter dan *Dense Layer*. ReLU merupakan fungsi aktivasi yang digunakan pada lapisan konvolusi dan lapisan tersembunyi pada *fully-connected layer*, dimana parameter ini akan melakukan perubahan masukan (*input*) menjadi nol jika masukan bernilai negatif, jika masukan positif maka nilai keluaran (*output*) akan sama dengan nilai masukan.

Pelatihan model pada penelitian ini terdiri dari beberapa tahapan yaitu:

1. *Compile Model*. Arsitektur model yang sudah dibuat akan disusun atau di-*compile* sehingga model menjadi siap untuk melakukan proses *training* menggunakan *pseudocode*:

```

# Compile Model
model.compile(loss = 'categorical_crossentropy',
              optimizer='Adam', metrics = ['accuracy'])

```

Variabel *loss* merupakan metode pengukuran nilai *loss* berdasarkan pada nilai tertentu. Arsitektur pada penelitian ini menggunakan *Optimizer Adam* untuk memperbarui *weight network* secara iteratif berdasarkan data *training*. Dalam mendapatkan performa dari proses pelatihan menggunakan parameter *accuracy* sebagai nilai matriksnya. Dikarenakan *flow data train* yang dibuat *class_mode*-nya merupakan *categorical*, maka untuk parameter *loss*-nya menggunakan *categorical_crossentropy*.

2. Penentuan *Callback*. *Callback* merupakan fungsi yang akan diterapkan pada tahapan tertentu dalam pelatihan model seperti menghentikan proses pelatihan saat mencapai akurasi tertentu, menyimpan model secara otomatis saat *epoch* tertentu, dan menyesuaikan tingkat pelatihan dari waktu ke waktu. Selain itu *callback* juga dapat menghindari *overfitting* atau *underfitting* Proses *callback* menggunakan *pseudocode* :

```

checkpointer = ModelCheckpoint(
    filepath='saved_models/model_vgg16.hdf5',
    monitor='val_accuracy', mode='max',
    verbose=1, save_best_only=True)
early_stopping = EarlyStopping(
    monitor='val_loss', mode='min',
    verbose=1, patience=3)
reduce_lr = ReduceLROnPlateau(
    monitor='val_loss', factor=0.2,
    patience=2, min_lr=0.001)
callbacks=[early_stopping, reduce_lr, checkpointer]

```

Variabel *ModelCheckpoint* merupakan metode yang digunakan untuk menyimpan model pada *epoch* tertentu. Penelitian ini menggunakan *EarlyStopping* untuk menghentikan proses pelatihan ketika performa model tidak sesuai hasil yang diinginkan, *ReduceLROnPlateau* yang akan mengubah *learning_rate* atau kecepatan pembelajaran model pada proses pelatihan. Parameter *callbacks* metode yang digunakan untuk mengontrol proses pelatihan model sesuai dengan parameter yang diinginkan. Penggunaan *callback* akan membuat pelatihan otomatis berhenti ketika *val_accuracy* tidak mengalami peningkatan yang signifikan, kemudian model akan otomatis tersimpan ketika *val_accuracy* mengalami peningkatan pada tiap *epoch* nya.

3. *Fitting Model*. Model yang sudah di *compile* dan sudah ditentukan *callback*-nya sudah dapat dilakukan proses training. Proses pelatihan dapat dilakukan dengan menggunakan fungsi *fitting* model dengan *pseudocode* :

```

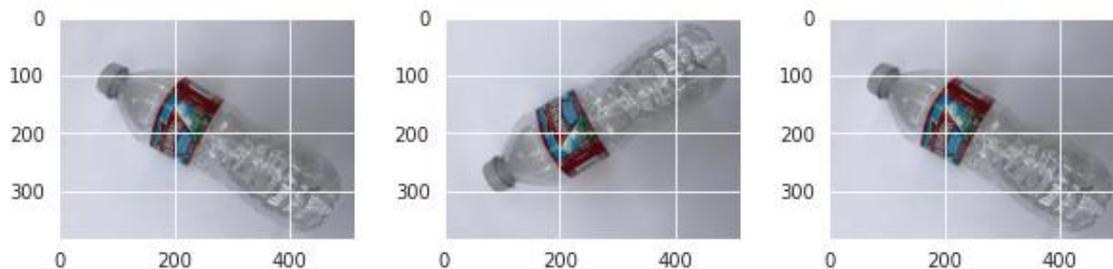
history = model.fit(train_generator,
    epochs = 40,
    validation_data = val_generator,
    callbacks=callbacks)

```

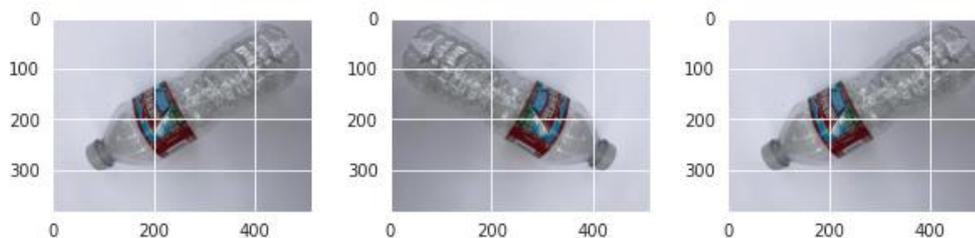
Parameter *model.fit* melakukan proses *fitting* model menggunakan *train_generator* yang merupakan data *generator* yang sudah diproses dan merupakan data yang digunakan pada proses pelatihan. Pengaturan *epoch* sejumlah 40 *epochs*.

HASIL DAN PEMBAHASAN

Pada tahap *pre-processing* data, setiap data dilakukan proses *rescale* dan augmentasi data untuk keperluan memperbanyak dan mengumpulkan *dataset* yang lebih bervariasi atau beragam. Dimensi dari data yang berupa citra sampah yang akan digunakan dalam proses pelatihan akan diubah menjadi 300x300 piksel. Dalam mendapatkan dataset yang beragam maka penelitian ini melakukan augmentasi untuk memperbanyak variasi dataset pada proses pelatihan model klasifikasi, seperti *vertical flip* yang merupakan salah satu proses augmentasi data yang digunakan untuk mengubah citra secara *vertical*. Contoh hasil dari proses *vertical flip* ditampilkan pada Gambar 2 dan Gambar 3 yang merupakan *Horizontal flip* untuk mengubah citra secara horizontal.



Gambar 2. Augmentasi Data *Vertical Flip*



Gambar 3. Augmentasi Data *Horizontal Flip*

Nilai *accuracy* menunjukkan bahwa proses pengujian dari model CNN yang telah dilatih dapat mengklasifikasikan jenis sampah dengan benar sebanyak 82.52% dari seluruh 1229 citra sampah yang berasal dari data *test*. Gambar 4 merupakan contoh hasil dari proses pengujian yang divisualisasikan.



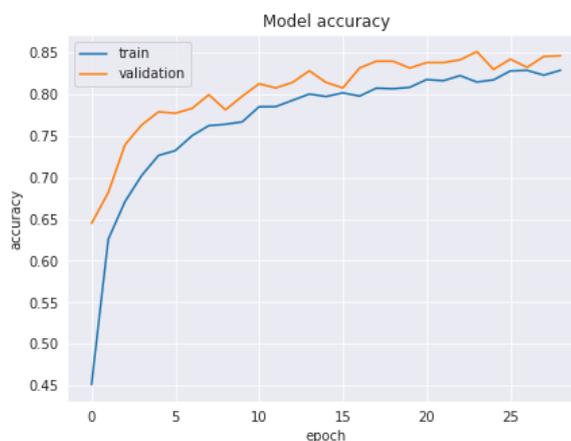
Gambar 4. Contoh Hasil Pengujian

Pelatihan model dilakukan sebanyak 40 *epoch*. Seperti dapat dilihat pada Tabel 3, penggunaan parameter *callback* pada proses pelatihan akan otomatis terhenti ketika *val_accuracy* tidak mengalami peningkatan yang signifikan, saat *epoch* ke-25 sampai *epoch* ke-29 nilai *val_accuracy* tidak mengalami peningkatan.

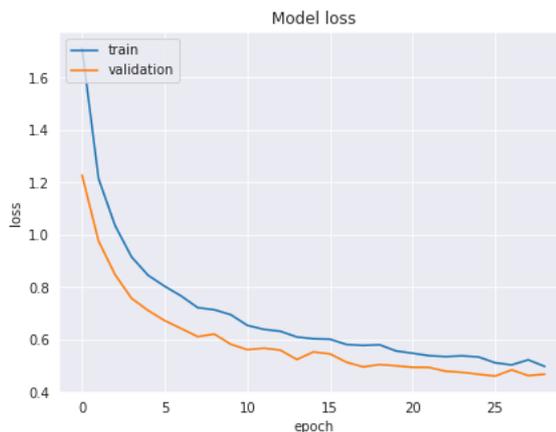
Tabel 3. Hasil Pelatihan Model

Epoch Ke-	Waktu Pelatihan (detik)	Nilai Accuracy	Nilai Loss	Val Accuracy	Val Loss	Status Callbacks
1	158	0.4511	1.7120	0.6447	1.2276	<i>val_accuracy improved, saving model</i>
2	139	0.6257	1.2139	0.6817	0.9758	<i>val_accuracy improved, saving model</i>
....						
23	138	0.8224	0.5347	0.8413	0.4788	<i>val_accuracy improved, saving model</i>
24	138	0.8146	0.5376	0.8512	0.4747	<i>val_accuracy improved, saving model</i>
25	138	0.8173	0.5333	0.8298	0.4673	<i>val_accuracy did not improve</i>
26	138	0.8280	0.5110	0.8421	0.4601	<i>val_accuracy did not improve</i>
27	138	0.8287	0.5030	0.8322	0.4841	<i>val_accuracy did not improve</i>
28	138	0.8229	0.5224	0.8454	0.4621	<i>val_accuracy did not improve</i>
29	138	0.8289	0.4972	0.8462	0.4675	<i>val_accuracy did not improve, early stopping</i>

Pelatihan model CNN untuk klasifikasi jenis sampah menggunakan algoritma VGG-16 menghasilkan nilai *accuracy* sebesar 82.89%, dan nilai *val_accuracy* sebesar 84.62%. Hasil ujicoba menunjukkan bahwa model yang dibangun menghasilkan loss sejumlah 17.11%, hal ini merupakan suatu kondisi ketika model tidak bisa melakukan prediksi yang tepat jika diberikan dataset serupa, menangkap *noise* yang seharusnya diabaikan dan menyebabkan tingkat akurasi rendah dan nilai *loss* yang tinggi. Visualisasi kedalam bentuk grafik untuk nilai *accuracy* dan *loss* selama pelatihan model klasifikasi jenis sampah dapat dilihat pada Gambar 5 dan Gambar 6.

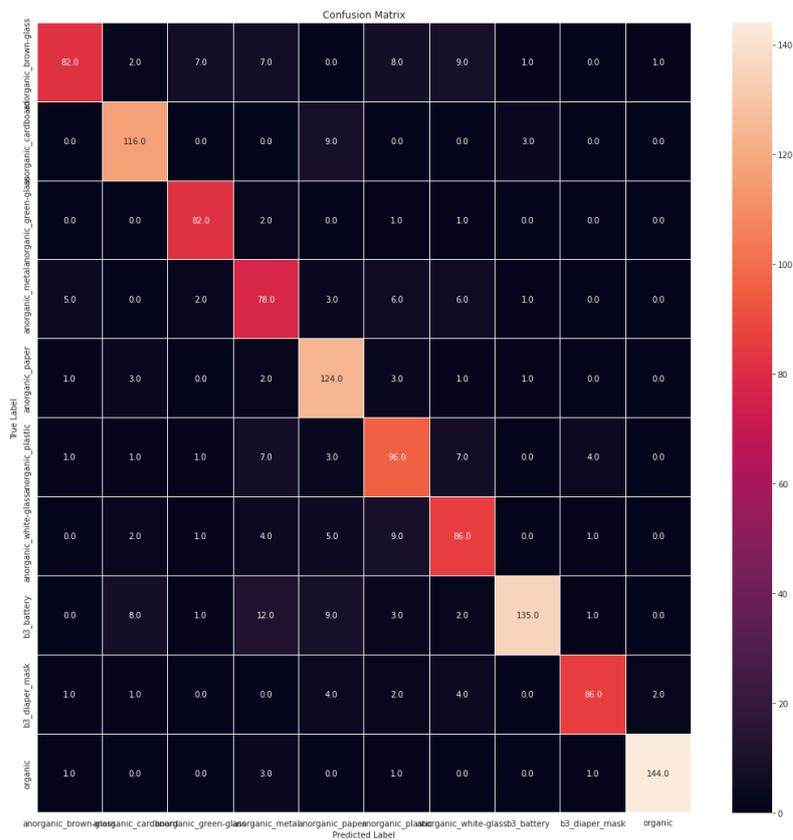


Gambar 5. Grafik Train dan Validation Accuracy



Gambar 6. Grafik *Train and Validation Loss*

Evaluasi model CNN selain dilihat dari nilai akurasi dapat dilihat juga dari hasil *confusion matrix* seperti dapat dilihat pada Gambar 7.



Gambar 7. Hasil *Confusion Matrix*

Hasil dari *confusion matrix* juga dapat digunakan untuk menghitung nilai ukuran evaluasi model yang lainnya, seperti *precision*, *recall/sensitivity*, *f1-score*, dan *support* seperti dapat dilihat pada Tabel 4.

Tabel 4. Nilai Ukuran Evaluasi Model Lainnya

Target Class	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>	<i>Support</i>
anorganic_brown-glass	0.700855	0.901099	0.788462	91.000000
anorganic_cardboard	0.906250	0.872180	0.888889	133.000000
anorganic_green-glass	0.953488	0.872340	0.911111	94.000000
anorganic_metal	0.772277	0.678261	0.722222	115.000000
anorganic_paper	0.918519	0.789809	0.849315	157.000000
anorganic_plastic	0.800000	0.744186	0.771084	129.000000
anorganic_white-glass	0.796296	0.741379	0.767857	116.000000
b3_battery	0.789474	0.957447	0.865385	141.000000
b3_diaper_mask	0.860000	0.924731	0.891192	93.000000
organic	0.960000	0.979592	0.969697	147.000000
accuracy	0.846217	0.846217	0.846217	0.846217
macro avg	0.845716	0.846102	0.842521	1216.000000
weighted avg	0.851104	0.846217	0.845396	1216.000000

KESIMPULAN DAN SARAN

Berdasarkan hasil ujicoba klasifikasi jenis sampah menggunakan metode *Convolutional Neural Network* (CNN) dengan arsitektur VGG-16 terhadap citra sampah, dapat diambil beberapa kesimpulan antara lain: Model CNN berhasil dibentuk menggunakan algoritma CNN dengan arsitektur VGG-16 sebagai metode *transfer learning*, yang kemudian dimodifikasi dengan penambahan beberapa lapisan diantaranya *pooling layer* dengan *GlobalAveragePooling2D*, *dense layer* dengan aktivasi ReLU, dan *dense layer* dengan aktivasi *softmax* sebagai lapisan *full connected layer*. Pelatihan model menggunakan data *train* dan data *validation*. Dimana data *train* digunakan untuk pelatihan model, sedangkan data *validation* digunakan untuk proses validasi model sehingga dapat melihat performa dari model CNN yang telah dibuat. Pelatihan dilakukan dengan menerapkan *callback* pada prosesnya, sehingga pelatihan akan otomatis terhenti ketika *val_accuracy* tidak mengalami peningkatan yang signifikan yaitu pada saat *29 epoch*. Pengujian dilakukan dengan tujuan untuk mengetahui tingkat keakuratan model CNN yang telah dibangun menggunakan data yang belum dikenali oleh model yaitu data *test* sebanyak 1229 data berbentuk citra sampah dan menghasilkan nilai *accuracy* sebesar 82.52%. Terdapat loss sejumlah 17.11%, hal ini diakibatkan kurangnya jumlah variasi pada data pelatihan. Tingkat keakuratan pengujian dipengaruhi oleh beberapa faktor seperti banyaknya jumlah data yang digunakan sebagai data *train* untuk model yang digunakan, seberapa banyak model data dilatih dalam mengenali setiap pola citra sampah yang ada pada data *train*, banyak atau sedikitnya layer yang digunakan pada *feature learning* ataupun *classification layer* dalam arsitektur model yang akan digunakan, pembagian rasio data yang akan digunakan pada proses *training*, *validation*, sampai *testing* dan faktor-faktor lainnya.

Pengembangan lebih lanjut dalam membentuk model klasifikasi jenis sampah dapat dilakukan dengan menggunakan *dataset* dengan jumlah citra yang lebih banyak lagi agar model dapat mempelajari fitur dari citra dengan lebih baik dan variasi lebih luas misalnya dengan melakukan augmentasi untuk mendapatkan variasi data. Data yang digunakan pada penelitian selanjutnya diharapkan menggunakan data yang diambil dengan latar belakang yang alami agar dapat digunakan secara praktikal untuk mengidentifikasi citra sampah yang diambil langsung tanpa perlu dilakukan *preprocessing* dan ekstraksi fitur yang rumit.

DAFTAR PUSTAKA

- Amasuomo, E., & Baird, J. (2016). The Concept of Waste and Waste Management. *Journal of Management and Sustainability*, 6(4), 88-96. <https://doi.org/10.5539/jms.v6n4p88>
- Aprilia, A. (2021). Waste Management in Indonesia and Jakarta: Challenges and Way Forward. *23rd ASEF Summer University, October*, 1-18. https://asef.org/wp-content/uploads/2022/01/ASEFSU23_Background-Paper_Waste-Management-in-Indonesia-and-Jakarta.pdf
- Classification, G. (2018). *No Title*. <https://www.kaggle.com/datasets/asdasdasdasdas/garbage-classification>
- Fang, B., Yu, J., Chen, Z., Osman, A. I., Farghali, M., Ihara, I., Hamza, E. H., Rooney, D. W., & Yap, P. S. (2023). Artificial intelligence for waste management in smart cities: a review. In *Environmental Chemistry Letters* (Vol. 21, Issue 4). Springer International Publishing. <https://doi.org/10.1007/s10311-023-01604-3>
- Ibnul Rasidi, A., Pasaribu, Y. A. H., Ziqri, A., & Adhinata, F. D. (2022). Klasifikasi Sampah Organik dan Non-Organik Menggunakan Convolutional Neural Network. *Jurnal Teknik Informatika Dan Sistem Informasi*, 8(1), 142-149. <https://doi.org/10.28932/jutisi.v8i1.4314>
- Ichtiakhiri, T. H., & Sudarmaji. (2015). Pengelolaan Lumbang B3 dan Keluhan Kesehatan Pekerja Di PT. INKA (Persero) Kota Madiun. *Jurnal Kesehatan Lingkungan*, 08(1), 118-127.
- Kohsasih, K. L., Rizky, M. D. A., Fahriyani, T., Wijaya, V., & Rosnelly, R. (2021). Analisis Perbandingan Algoritma Convolutional Neural Network dan Algoritma Multi-layer Perceptron Neural dalam Klasifikasi Citra Sampah. *Jurnal TIMES*, 10(2), 22-28.
- Malik, M., Sharma, S., Uddin, M., Chen, C. L., Wu, C. M., Soni, P., & Chaudhary, S. (2022). Waste Classification for Sustainable Development Using Image Recognition with Deep Learning Neural Network Models. *Sustainability (Switzerland)*, 14(12), 1-18. <https://doi.org/10.3390/su14127222>
- Mittal, I., Tiwari, A., Rana, B., & Singh, P. (2020). *Trash Classification: Classifying garbage using Deep Learning*. 11(July), 61–68. www.jespublication.com
- Parkar, S., Mulukh, R., Narhari, G., & Kulkarni, S. (2021). An Insight Into Treatment, Reuse, Recycle and Disposal of Biodegradable and Non-biodegradable Solid Waste. *SSRN Electronic Journal*. <https://doi.org/10.2139/ssrn.3867475>
- Poudel, S., & Poudyal, P. (2022). Classification of Waste Materials using CNN Based on Transfer Learning. *ACM International Conference Proceeding Series*, 29-33. <https://doi.org/10.1145/3574318.3574345>
- Satispi, E., & Aziz Samudra, A. (2022). Study of Policy Implementation: Strategy of COVID-19 Plastic Waste Management in Indonesia. *Journal of Public Policy and Administration*, 6(4), 155-164. <https://doi.org/10.11648/j.jpaa.20220604.11>
- Stephen, Raymond, & Santoso, H. (2019). Aplikasi Convolution Neural Network untuk Mendeteksi Jenis-Jenis Sampah. *Explore – Jurnal Sistem Informasi Dan Telematika*, 10(2), 122-132.
- Hutamaputra, W., Krisnabayu, R. Y., Mawarni, M., Yudistira, N., & Bachtiar, F. A. (2022). Perbandingan Convolutional Neural Network VGG16 dan ResNet34 pada Sistem Klasifikasi Sampah Botol. *Jurnal Teknologi Dan Sistem Komputer*.

<https://doi.org/https://doi.org/10.14710/jtsiskom.2021.14045>

Wulandari, S. A., Ma'ruf, M., Priyatno, A. R., Halimun, N., Abdulah, Z. M., & Amartiwi, U. (2023). DjunkGo: A Mobile Application for Trash Classification with VGG16 Algorithm. *GMPI Conference Series*, 2(May 2022), 67-72. <https://doi.org/10.53889/gmpics.v2.175>

How to cite:

Sutanty, E., Maukar, M., Astuti, D. K., Handayani, H. (2023). Penerapan Model Arsitektur VGG16 untuk Klasifikasi Jenis Sampah. *DECODE: Jurnal Pendidikan Teknologi Informasi*, 3(2), 407-419. <http://dx.doi.org/10.51454/decode.v3i2.331>