

## ***SISTEM PENGENALAN BOTOL PLASTIK BERDASARKAN LABEL MEREK MENGGUNAKAN FASTER-RCNN***

**Arief Setyo Nugroho<sup>1</sup>, Rusydi Umar<sup>2</sup>, Abdul Fadlil<sup>3</sup>**

Magister Teknologi Informasi, Universitas Ahmad Dahlan

Fakultas Teknologi Industri, Universitas Ahmad Dahlan

---

### **Informasi Makalah**

Dikirim, 23 September 2020

Direvisi, 12 Oktober 2020

Diterima, 25 Oktober 2020

---

### **Kata Kunci:**

*Botol Plastik*

*Convolutional Neural Network*

*Faster-RCNN*

*Label Merek*

*Tensorflow*

---

### **Keyword:**

Plastic Bottles

Convolutional Neural Network

Faster-RCNN

Brand Label

Tensorflow

---

### **INTISARI**

Penumpukan botol plastik saat ini sudah tidak terkendali sehingga mengakibatkan polusi pada lingkungan. Sampah botol plastik saat ini dapat ditukar dengan imbalan yang beragam. Sehingga proses sortir botol plastik dapat dilakukan untuk memilih sampah botol plastik. Pada penelitian ini dibuat sistem yang dapat mengenali dan mengklasifikasi botol plastik berdasarkan label merek dengan 5 kelas berukuran sedang atau 600ml. Metode yang akan digunakan adalah teknik pengolahan citra dengan menggunakan *Convolutional Neural Network* dengan *Tensorflow* dan model data *Faster-RCNN*. Penelitian dibagi menjadi 3 bagian yaitu *pre-processing*, *training*, dan *testing*. Pengujian dilakukan dengan menampilkan hasil dari proses bagian yang akan dilakukan serta menampilkan hasil akurasi. Berdasarkan dari hasil pengujian sistem dapat mengenali objek dengan baik dengan akurasi sebesar 87,12%

---

### **ABSTRACT**

Plastic bottles waste is currently out of control, causing pollution to the environment. Plastic bottles can be exchanged for various rewards. So that the plastic bottles sorting process can be used to select plastic bottles waste. In this research, a system was created that could analyze and classify plastic bottles based on brand labels with 5 classes of medium size or 600ml. The method used is an image processing technique using Convolutional Neural Network with Tensorflow and a Faster-RCNN model. The research is divided into 3 parts, pre-processing, training, and testing. Testing on research is the process classification object and the accuracy of the system. Based on the test, result of the system can recognize objects with an accuracy of 87.12%

---

### **Korespondensi Penulis:**

Arief Setyo Nugroho

Magister Teknologi Informasi

Universitas Ahmad D

JL. Prof. DR. Soepomo S.H., No. 80 Warungboto, Kec. Umbulharjo, Yogyakarta, D.I. Yogyakarta, 55164

Email: [arief1908048023@webmail.uad.ac.id](mailto:arief1908048023@webmail.uad.ac.id)

---

## **1. PENDAHULUAN**

Penumpukan sampah botol plastik yang tidak terkendali dapat mengakibatkan polusi pada lingkungan. Peningkatan penggunaan botol plastik menjadi salah satu penyebab dimana saat ini telah banyak perusahaan minuman dengan berbagai merek yang menggunakan plastik sebagai bahan baku utama dalam membuat botol kemasan. Botol plastik membutuhkan waktu yang sangat lama untuk bisa terurai (1). Sampah botol plastik dapat dikurangi dengan salah satu caranya adalah daur ulang. Hasil dari daur ulang botol plastik dapat dimanfaatkan untuk membuat produk baru yang berbahan dasar plastik (2). Selain dibuat produk baru sampah botol plastik juga dapat ditukar dengan sebuah imbalan seperti yang dilakukan oleh perusahaan Danone

yang mengembangkan *Smart Drop Box* (3). Ada pula penelitian yang dilakukan oleh Handoko dkk, yang membuat mesin penukaran limbah botol kemasan dengan tiket yang dapat ditukar dengan mata uang (4). Penelitian tersebut menggunakan sensor *IR Obstacle Avoidance* untuk mendeteksi adanya botol dalam mesin. Namun metode tersebut hanya dapat mendeteksi ukuran botol plastik. Pendeteksian botol juga dapat dilakukan dengan sensor warna RGB TCS3200 (5).

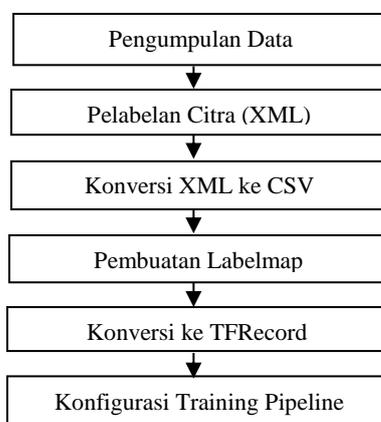
Pada penelitian ini akan dikembangkan sebuah metode yang dapat mengidentifikasi botol plastik dengan menggunakan teknik pengolahan citra (6). Identifikasi botol plastik dengan pengolahan citra sudah banyak dikembangkan seperti penelitian yang dilakukan oleh Wang dkk dengan melakukan klasifikasi daur ulang botol plastik yang ditentukan dengan warna dan posisi (7). Botol plastik juga memiliki bentuk warna yang berbeda sesuai dengan label merek masing-masing. Pengaruh penggunaan sebuah label merek sangat berpengaruh terhadap banyaknya penggunaan dan pembelian produk botol plastik (8). Sehingga pengenalan botol plastik berdasarkan merek dapat dijadikan objek klasifikasi karena identifikasi botol plastik berdasarkan merek tidak mudah dikenali jika hanya menggunakan sensor. Pengenalan merek sudah pernah dikembangkan oleh Alireza dkk yang mendeteksi logo dengan metode PPA (*Piece-Wise Painting Algoritm*) (9). Selain itu pengidentifikasian sebuah label merek juga dapat dilakukan dengan menggunakan metode *training data* (10).

Proses pengklasifikasian citra dapat dilakukan dengan berbagai macam metode salah satunya adalah CNN (*Convolutional Neural Network*). CNN merupakan salah satu jenis *Deep Learning* yang biasa diimplementasikan pada pengolahan citra (11). Beberapa penelitian yang sudah dilakukan menggunakan CNN salah satunya adalah penelitian yang dilakukan oleh Afiandi dkk yang mengimplementasikan CNN untuk mengklasifikasi pembalap MotoGP menggunakan GPU (12). *Deep Learning CNN* memiliki beberapa jenis *framework* yang dapat digunakan seperti *Tensorflow* yang dimiliki oleh *Google*, *Caffe2* yang dimiliki *Facebook*, *Microsoft* dengan CNTK serta beberapa jenis *framework* yang lain seperti *Theano* dan *PyTorch*.

Pada penelitian ini dibuat sebuah sistem yang dapat mengklasifikasi botol plastik berdasarkan merek menggunakan CNN. *Framework* yang akan digunakan pada penelitian ini adalah *Tensorflow* yang dibuat oleh *Google*. *Tensorflow* merupakan sebuah *Machine Learning* yang beroperasi pada skala yang besar menggunakan data *flow graph* untuk melakukan perhitungan dan operasi yang sama secara bersamaan (13). Pengenalan botol plastik menggunakan *Tensorflow* sudah pernah dilakukan oleh Valentina dkk namun hanya dapat mengenali botol plastik (14). Model yang akan digunakan sebagai media pendukung dalam pembelajaran mesin untuk mengenali botol plastik berdasarkan merek adalah model *Faster R-CNN* dengan sebuah *inception V2* yang akan mengenali 5 kelas yaitu botol plastik dengan label merek *Aqua*, *Sprite*, *Fanta*, *Coca-Cola*, dan *Minute-Maid Pulpy* yang berukuran sedang atau 600ml. Model ini dipilih karena memiliki nilai akurasi yang baik dari pada model yang lainnya (15). Sehingga diharapkan penelitian ini dapat menjadi metode yang dapat digunakan dalam identifikasi sampah botol plastik.

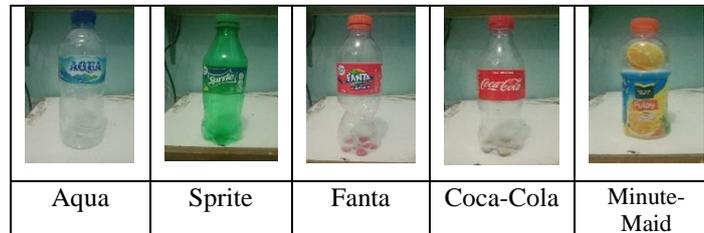
## 2. METODE

Penelitian ini akan menggunakan metode CNN dengan *framework* dari *Tensorflow* dengan model *Faster-RCNN*. Model ini dipilih karena dapat menghasilkan performa yang cepat dan akurat dibandingkan dengan model yang lain (16). Tahapan yang dilakukan pada penelitian ini adalah *Pre-processing*, *Training*, dan *Testing*. *Pre-processing* merupakan tahapan pertama dalam penelitian yang akan dilakukan seperti pada Gambar 1.



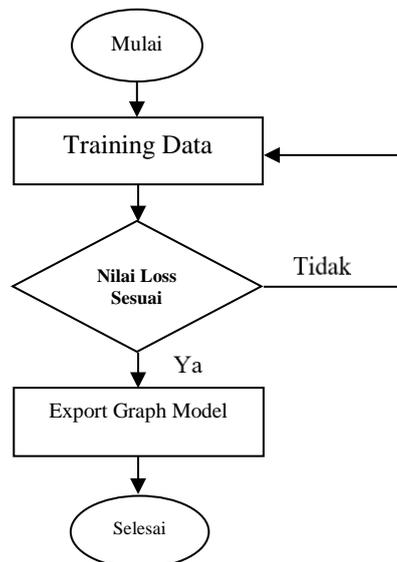
Gambar 1. *Pre-processing Faster-RCNN*

Pada bagian *Pre-processing* terdapat beberapa tahapan yang pertama adalah pengumpulan data. Proses pengumpulan data dilakukan dengan mengambil citra botol plastik berdasarkan label merek menggunakan kamera *Smartphone*. Citra botol plastik akan dibagi menjadi 5 kelas dengan berbagai label merek seperti pada Gambar 2. Merek yang digunakan yaitu *Aqua*, *Sprite*, *Fanta*, *Coca-Cola*, dan *Minute-Maid Pulpy*. Jumlah citra yang digunakan dalam penelitian ini dibagi menjadi 2 *folder* yaitu sebanyak 550 citra yang terbagi menjadi 100 citra untuk setiap merek botol plastik dan 50 campuran serta data *testing* sebanyak 190 citra dari 5 kelas yang digunakan dengan resolusi 2352 x 4160 pixel.



Gambar 2. Citra data botol plastik

Proses selanjutnya adalah pelabelan citra dengan menggunakan *software LabelImg.exe*. Pelabelan citra digunakan untuk memberikan informasi letak citra yang diinginkan dan akan disimpan dalam bentuk *.xml*. Pembuatan label dilakukan dengan cara memberikan batas kotak pada citra yang diinginkan dan diberikan penamaan kelas pada setiap citra. Selain itu pemberian label dinilai sangat penting untuk menghasilkan data dari proses *training* yang lebih akurat (17). Hasil pelabelan selanjutnya akan dirubah menjadi *csv* sebelum data dikonversi kembali ke *TfRecord*. Konversi *xml* ke *csv* akan dibagi menjadi 2 file yaitu *train* dan *test*. Namun sebelumnya akan dibuat sebuah konfigurasi penamaan atau yang biasa disebut dengan konfigurasi *Labelmap*. *Labelmap* digunakan untuk memberikan penamaan data agar bisa didefinisikan. Kemudian data *csv* akan dikonversi menjadi data yang dapat dibaca oleh *Tensorflow* yaitu menggunakan *TfRecord*. Langkah selanjutnya adalah membuat *file* konfigurasi yang nantinya akan digunakan untuk melakukan konfigurasi dari model *training*. Model *Tensorflow* yang akan digunakan adalah *Faster-RCNN Inception V2*. *Faster-RCNN* dinilai memiliki tingkat akurasi yang baik untuk identifikasi citra (18). Setelah semua tahap *pre-processing* selesai selanjutnya adalah proses *training* model data dengan tahapan seperti pada Gambar 3.



Gambar 3. Proses Training Data

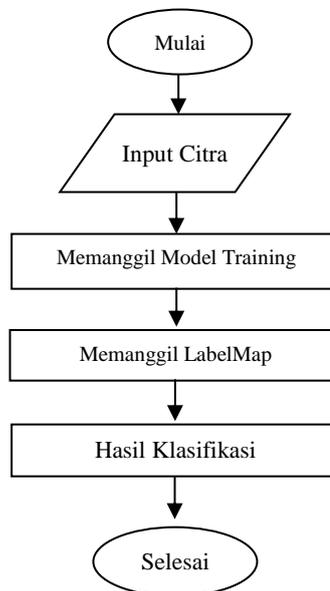
Proses *training* data akan diimplementasikan menggunakan *Tensorflow* dengan versi 1.15 dengan bahasa pemrograman *Python* dengan *software Anaconda*. *Tensorflow* memiliki dua metode *training* data yaitu menggunakan CPU dan GPU. Menurut John Lawrence dkk proses *training* data menggunakan GPU merupakan opsi yang paling baik karena memiliki performa yang bagus (19). Detail *hardware* yang akan digunakan pada penelitian ini yaitu *Windows 10 64 bit*, *Processor Intel Core i5 3317U*, *GPU NVIDIA GeForce*

740M dan RAM 8 GB. Selanjutnya adalah proses *training* data *Faster-RCNN* yang digunakan untuk melatih data citra. Saat proses *training* berlangsung akan ditampilkan hasil perhitungan dari nilai *loss* serta waktu *training* yang diperlukan sampai mendapatkan nilai *loss* yang diinginkan. Pada penelitian ini target nilai *loss* dari *training* data yang dilakukan adalah kurang atau sama dengan 0.1. Perhitungan *loss* pada saat *training* data dapat menggunakan Persamaan 1.

$$L(\{pi\}, \{ti\}) = \frac{1}{N_{cls}} \sum_i L_{cls}(pi, pi^*) + \lambda \frac{1}{N_{reg}} \sum_i Pi * L_{reg}(ti, ti^*) \quad (1)$$

Dimana ( $L_{cls}$ ) merupakan *Classification Loss*, ( $L_{reg}$ ) nilai *Regression Loss*, ( $pi$ ) nilai probabilitas yang diprediksi, ( $ti$ ) nilai koordinat 4k. Nilai  $pi^*$  dan  $ti^*$  merupakan nilai label dasar kebenaran dan dasar kebenaran pada masing masing koordinat. Sedangkan ( $N_{cls}$ ) dan ( $N_{reg}$ ) adalah nilai Normal dengan keseimbangan dari  $\lambda$  (18).

Setelah proses *training* data mendapatkan nilai *loss* yang diinginkan selanjutnya adalah mengekstrak model data menjadi *Inference Graph* agar dapat digunakan untuk mengklasifikasi data citra. Pengujian data citra botol plastik akan dilakukan dengan menggunakan 10 sampel citra dari masing-masing merek. Tahapan proses pengujian data dengan menggunakan hasil *training* terlihat seperti pada Gambar 4.



Gambar 4. Proses *Testing* Data Citra

Proses pengujian data atau *testing* dilakukan dengan memasukkan citra uji berupa botol plastik. Kemudian sistem akan membaca model data hasil dari *training* sebelumnya dan selanjutnya diproses untuk mengenali botol plastik berdasarkan merek. Hasil dari proses *training* telah dikonversi menjadi *file protobuf* (20). Untuk mendapatkan definisi dari label prediksi yang sesuai maka perlu memanggil *labelmap* dan akan diukur nilai akurasi berdasarkan hasil prediksi klasifikasi menggunakan Persamaan 2.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (2)$$

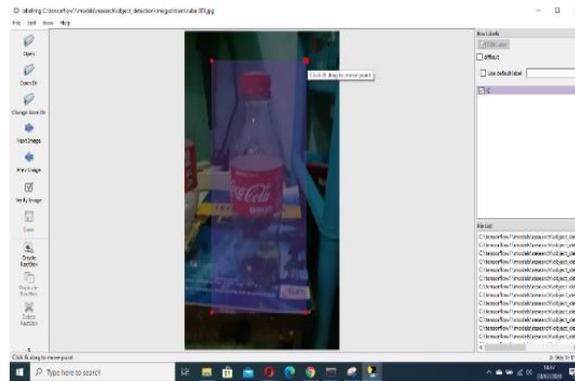
Akurasi klasifikasi diukur berdasarkan nilai TP (*True Positive*) yang merupakan nilai kebenaran dari sebuah prediksi, nilai TN (*True Negative*) atau nilai prediksi yang ditolak dalam keadaan *positive*. Kemudian nilai FP (*False Positive*) atau nilai kelas yang teridentifikasi salah sedangkan FN (*False Negative*) yaitu nilai kelas yang ditolak dalam keadaan *negative* (21).

### 3. HASIL DAN PEMBAHASAN (10 PT)

Hasil dari penelitian ini akan dibagi menjadi beberapa bagian yaitu saat proses *pre-processing*, *training* dan hasil *testing*. Pada pengujian pertama akan dimulai dengan menampilkan hasil dari *pre-processing*

### 3.1 Pre-processing

Penelitian ini dimulai dengan mengumpulkan data citra botol plastik dengan label merek yang berbeda menggunakan kamera *smartphone*. Setelah data terkumpul selanjutnya dilakukan proses pelabelan citra menggunakan *software LabelImg.exe*. Pada proses pelabelan, citra yang diseleksi adalah citra botol plastik dengan memberikan tanda kotak serta memberikan nama label yang kemudian disimpan dalam bentuk *.xml* seperti pada Gambar 5.



Gambar 5. Proses Pelabelan Citra

Setelah seluruh data citra diberikan label, selanjutnya adalah proses konversi *xml* ke *csv*. Pada model *Faster R-CNN* sudah disediakan program konversi untuk mengubah *file xml* ke *csv* yaitu *xml\_to\_csv.py*. Tahap selanjutnya adalah pembuatan *file Labelmap* menggunakan *text editor* dalam bentuk *.pbtxt* yang terlihat pada Gambar 6.

```

1  item {
2    id: 1
3    name: 'A'
4  }
5
6  item {
7    id: 2
8    name: 'S'
9  }
10
11 item {
12  id: 3
13  name: 'F'
14  }
15
16 item {
17  id: 4
18  name: 'C'
19  }
20
21 item {
22  id: 5
23  name: 'M'
24  }
25

```

Gambar 6. Isi *Labelmap*

Isi *Labelmap* terdiri dari nama dari setiap kelas dengan menggunakan simbol untuk menandai setiap botol plastik yang terdeteksi. Simbol pada *Labelmap* terlihat pada Tabel 1.

Tabel 1. Daftar Simbol Botol Plastik

Nama Label	Simbol
Aqua	A
Coca-Cola	C
Fanta	F
Minute-Maid Pulpy	M
Sprite	S

Setelah *Labelmap* dibuat selanjutnya adalah proses konfigurasi ke *TfRecord*. Didalam model telah pula disediakan program untuk mengkonfigurasi *TfRecord* dengan nama kode *generate\_tfrecord.py*. Pada program tersebut harus dilakukan perubahan pada baris 31 dengan isi seperti pada Gambar 7.

```

30 # TO-DO replace this with label map
31 def class_text_to_int(row_label):
32     if row_label == 'A':
33         return 1
34     elif row_label == 'S':
35         return 2
36     elif row_label == 'F':
37         return 3
38     elif row_label == 'C':
39         return 4
40     elif row_label == 'M':
41         return 5
42     else:
43         return 0
44

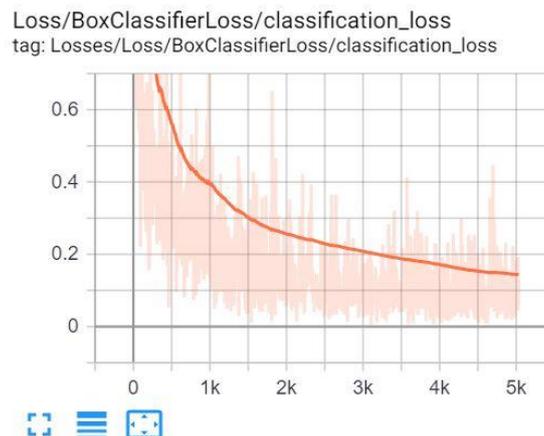
```

Gambar 7. Konfigurasi *TFRecord*

Isi dari konfigurasi *TFRecord* adalah mengisi *row-label* sesuai simbol pada *Labelmap* yang sudah dibuat. Setelah konfigurasi selesai kemudian dilanjutkan dengan melakukan konfigurasi *training*. Konfigurasi *training* terdiri dari definisi model dan parameter yang akan digunakan untuk *training*. Pada proses konfigurasi *training* digunakan program konfigurasi dari *Faster-RCNN* dengan nama file *faster\_rcnn\_inception\_v2\_pets.config*. Setelah seluruh konfigurasi dilakukan langkah selanjutnya adalah proses *training* data.

### 3.2 Training data

Pada proses *training* data akan menggunakan program bawaan dari model *Tensorflow* yaitu *train.py*. Pada saat proses *training* berlangsung akan terlihat hasil dari nilai *loss* dari *training* data berdasarkan Persamaan (1). Saat *training* dilakukan menggunakan *Faster-RCNN-Inception-V2*, dimulai dengan nilai 5.0 dan berjalan hingga nilai *training* mencapai kurang atau sama dengan 1.0. Pada penelitian ini nilai *training* dilakukan sebanyak 5033 *step* dengan nilai *loss* paling kecil sebesar 0.8. Namun membutuhkan waktu yang cukup lama dalam proses *training* yaitu selama lebih dari 48 jam. Hal tersebut masih kurang efektif jika komponen *hardware* masih digunakan untuk *training* data dengan *Tensorflow*. Sedangkan saat komponen *hardware* memadai proses *training* akan semakin cepat dan penggunaan data untuk *training* akan semakin banyak. Saat proses *training* berlangsung *progress training* akan ditampilkan pada fitur yang dimiliki oleh *Tensorflow* yaitu *Tensorboard*. Fitur ini akan menampilkan informasi dan grafik saat proses *training* berlangsung. Grafik *progress loss* dapat dilihat pada Gambar 8 dengan melihat nilai *loss* pada saat klasifikasi.

Gambar 8. Grafik *Loss Box Classifier*

*Training* akan tersimpan otomatis pada setiap menitnya atau biasa disebut dengan *checkpoint*. Setelah *training* selesai dilakukan selanjutnya adalah konversi hasil *training* ke *Inference graph*.

### 3.3 Pengujian Data

Setelah proses *training* selesai dilakukan maka selanjutnya adalah proses pengujian data menggunakan program *Object\_detection\_image.py* yang disediakan oleh model. Dalam penelitian ini akan

diukur akurasi deteksi yang dihasilkan oleh sistem terhadap hasil deteksi citra klasifikasi. Hasil pengujian klasifikasi dan identifikasi citra botol plastik dapat dilihat pada Tabel 2. Pada citra hasil deteksi seperti Gambar 9 terlihat saat botol plastik terdeteksi maka citra botol akan ditandai dengan kotak disekelilingnya dan akan ditampilkan nama kode dan nilai akurasi.



Gambar 9. Hasil deteksi citra

Tabel 2. Hasil Pengujian Klasifikasi Botol Plastik

Nama Botol Plastik	Jumlah Data	Jumlah Terdeteksi	Akurasi (%)
Aqua	10	9	78,5
Coca-cola	10	10	97
Fanta	10	7	65,9
Sprite	10	10	98,3
Minute-Maid Pulpy	10	10	95,9

Berdasarkan hasil dari pengujian mendapatkan nilai total rata-rata akurasi berdasarkan Persamaan 2 adalah 87.12%. Dari hasil tersebut terlihat hasil deteksi botol pada merek Fanta mendapatkan akurasi yang lebih kecil dari data pengujian botol plastik yang lainnya. Hal tersebut bisa dikarenakan proses training data yang kurang atau jumlah data *training* yang kurang banyak. Selain itu pula bentuk dan warna dari botol merek Fanta dan Coca-cola yang hampir mirip.

#### 4. KESIMPULAN

Berdasarkan hasil dari penelitian ini sistem dapat mendeteksi dan mengklasifikasi botol plastik berdasarkan label merek dengan baik. Sistem berhasil mendapatkan nilai akurasi sebesar 87.12%. Sehingga sistem dapat digunakan sebagai metode yang bisa diimplementasikan untuk sortir sampah botol plastik. Pada penelitian selanjutnya penambahan jumlah data dan jumlah *step* dalam *training* dapat diperbanyak agar mendapatkan hasil yang lebih baik.

#### DAFTAR PUSTAKA

- [1] Budianto A. Pirolisiss Botol Plastik Bekas Minuman Air Mnieral Jenis Pet Menjadi Fuel. In: Seminar Nasional Sains dan Teknologi Terapan V [Internet]. 2017. p. 201–6. Available from: <http://conference.itats.ac.id/index.php/sntekpan/2017/paper/view/156>
- [2] Dahlbo H, Poliakova V, Mylläri V, Sahimaa O, Anderson R. Recycling potential of post-consumer plastic packaging waste in Finland. Waste Manag [Internet]. 2018;71:52–61. Available from: <https://doi.org/10.1016/j.wasman.2017.10.033>
- [3] Qonaah S. Strategy Kampanye Gerakan #BijakBerplastik PT Danone Aqua Dalam Merayakan Hari Lingkungan Hidup Sedunia 2018. J Komun. 2019;10(1):48–55.

- [4] Handoko P, Hermawan H, Jaya S. Reverse Vending Machine Penukaran Limbah Botol Kemasan Plastik Dengan Tiket Sebagai Alat Tukar Mata Uang. In: Seminar Nasional Sains dan Teknologi 2018. 2018. p. 1–12.
- [5] Wiryadinata R, Hamaedi FA, Martiningsih W. Perancangan Sistem Penyortir Botol dengan Menggunakan Sensor Warna RGB TCS3200. Semin Nas Tek Elektro 2018. 2018;163–7.
- [6] Umar R, Riadi I, Miladiah M. Sistem Identifikasi Keaslian Uang Kertas Rupiah Menggunakan Metode K-Means Clustering. *TechnoCom*. 2018;17(2):179–85.
- [7] Wang Z, Peng B, Huang Y, Sun G. Classification for plastic bottles recycling based on image recognition. *Waste Manag* [Internet]. 2019;88:170–81. Available from: <https://doi.org/10.1016/j.wasman.2019.03.032>
- [8] Sari V, Suryoko S. Pengaruh Harga, Kualitas Produk Dan Citra Merek Terhadap Keputusan Pembelian Teh Siap Minum Dalam Kemasan Teh Botol Sosro (Studi Kasus Pada Mahasiswa S1 Fisip Undip Semarang). *J Ilmu Adm Bisnis S1 Undip*. 2017;6(3):453–64.
- [9] Alaei A, Delalandre M. A complete logo detection/recognition system for document images. In: *Proceedings - 11th IAPR International Workshop on Document Analysis Systems, DAS 2014*. 2014. p. 324–8.
- [10] Romberg S, Pueyo LG, Lienhart R, Van Zwol R. Scalable logo recognition in real-world images. *Proc 1st ACM Int Conf Multimed Retrieval, ICMR'11*. 2011;(February 2016).
- [11] Putra IWSE, Wijaya AY, Soelaiman R. Klasifikasi Citra Menggunakan Convolutional Neural Network ( Cnn ) Pada Caltech 101 Image Classification Using Convolution Neural Network ( Cnn ) on Caltech 101. *Inst Teknol Sepuluh Novemb*. 2016;5(1):1–76.
- [12] Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, et al. TensorFlow: A System for large-scale machine learning. In: *12th USENIX Symposium on Operating System Design and Implementation (OSDI'16)*. 2016. p. 265–83.
- [13] Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, et al. TensorFlow: A System for large-scale machine learning. In: *12th USENIX Symposium on Operating System Design and Implementation (OSDI'16)*. 2016. p. 265–83.
- [14] Valentina R, Rostianingsih S, Tjondrowiguno AN, Surabaya JS. Pengenalan Gambar Botol Plastik dan Kaleng Minuman Menggunakan Metode Convolutional Neural Network. *J Infra*. 2020;8(1).
- [15] Jiang H, Learned-Miller E. Face Detection with the Faster R-CNN. In: *2017 12th IEEE International Conference on Automatic Face Gesture Recognition (FG 2017)*. 2017. p. 650–7.
- [16] Ren S, He K, Girshick R, Sun J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans Pattern Anal Mach Intell*. 2017;39(6):1137–49.
- [17] Pella SI, Lami H. Disain sistem otomasi suhu ruangan pertemuan dengan penerapan teknik machine learning. *J Media Elektro*. 2019;8(2):116–20.
- [18] Alamsyah D, Fachrurrozi M. Faster R-CNN with inception v2 for fingertip detection in homogenous background image. *J Phys Conf Ser*. 2019;1196(1).
- [19] Lawrence J, Malmsten J, Rybka A, Sabol DA, Triplin K. Comparing tensorflow deep learning performance and experiences using CPUs via local PCs and cloud solutions. In: *Proceedings of Student-Faculty Research Day*. 2017. p. 1–6.
- [20] Hasma YA, Silfianti W. Implementasi Deep Learning Menggunakan Framework Tensorflow Dengan Metode Faster Regional Convolutional Neural Network Untuk Pendeteksian Jerawat. *J Ilm Teknol dan Rekayasa*. 2018;23(2):89–102.
- [21] Orozco C, Rebong C. Vehicular Detection and Classification for Intelligent Transportation System: A Deep Learning Approach Using Faster R-CNN Model. *Int J Simul Syst Sci Technol*. 2019;1–7.