

## DAFTAR PUSTAKA

1. Hendra, Y. Perbandingan Sistem Pengelolaan Sampah di Indonesia dan Korea Selatan: Kajian 5 Aspek Pengelolaan Sampah. *Aspirasi* **7**, 77–91 (2016).
2. Hasibuan, R. Analisis Dampak Limbah/Sampah Rumah Tangga Terhadap Pencemaran Lingkungan Hidup. *J. Ilm. Advokasi* (2016).
3. Harahap, R. D. Effect of Household Waste Viewed From the Aspect Environmental Conservation Biology in Housing Complex Graha Pertiwi Kel. Undo Kompas Kec. South Rantau. *Cahaya Pendidik*. **2**, 92–104 (20 ).
4. Harjanti, I. M. & Anggraini, P. Pengelolaan Sampah Di Tempat Pembuangan Akhir (TPA) Jatibarang, Kota Semarang. *J. Planol*. **17**, 185 (2020).
5. Pandey, L. M. S. & Shukla, S. K. An insight into waste management in Australia with a focus on landfill technology and liner leak detection. *J. Clean. Prod.* **225**, 1147–1154 (2019).
6. Sahil, J., Muhdar, M., Rohman, F. & Syamsuri, I. Waste management at Dufa Dufa subdistrict, City of Ternate (in Bahasa Indonesia). *BIOeduKASI* **4**, 478–487 (2016).
7. Yulida, N., Suwarni, A. & Sarto, S. Analysis of community behavior on garbage disposing in Batang Bakarek-Karek river basin of Padang Panjang. *Ber. Kedokt. Masy.* **32**, 373 (2016).
8. Zhao, Y., Zhang, Y., Cheng, R., Wei, D. & Li, G. An enhanced

- histogram of oriented gradients for pedestrian detection. *IEEE Intell. Transp. Syst. Mag.* **7**, 29–38 (2015).
9. Umam, K. & Negara, B. S. Deteksi Obyek Manusia Pada Basis Data Video Menggunakan Metode Background Subtraction Dan Operasi Morfologi. *J. CoreIT J. Has. Penelit. Ilmu Komput. dan Teknol. Inf.* **2**, 31 (2016).
  10. Sianturi, J., Rahmat, R. F. & Nababan, E. B. Sistem Pendeteksian Manusia untuk Keamanan Ruangan Human Detection System For Room Security Using Viola - Jones. *J. Informatics Telecommun. Eng.* **1**, 61–72 (2018).
  11. Elamin, M. Z. *et al.* Analysis of Waste Management in The Village of Disanah, District of Sreseh Sampang, Madura. *J. Kesehat. Lingkung.* **10**, 368 (2018).
  12. Nulhakim, L. Pemilahan Jenis Sampah Logam Dan Non-Logam Skala Kecil Secara Otomatis Berbasis Arduino( Smart Trash Can ). *J. Teknol. Inf. dan Komunikasi(FIKI)* **IX**, 59–66 (2019).
  13. Mustaqim, R. A. Pandangan Ulama Terhadap Image Processing Pada Astrofotografi Di BMKG Untuk Rukyatul Hilal. *Al-Marshad J. Astron. Islam dan Ilmu-Ilmu Berkaitan* **4**, 78–115 (2018).
  14. Darmawansyah, B., Zarkasih, A. & Prasetyo, A. P. P. Rancang Bangun Robot Wall Follower Menggunakan Sensor Kamera. 7–11 (2010).
  15. Khan, S., Rahmani, H., Shah, S. A. A. & Bennamoun, M. *A Guide to Convolutional Neural Networks for Computer Vision. Synthesis*

*Lectures on Computer Vision* vol. 8 (Morgan & Claypool, 2018).

16. Rasyid, M. R., Tahir, Z. & Syafaruddin, N. Digital Image Processing for Detecting Industrial Machine Work Failure with Quantization Vector Learning Method. *J. Pekommas* **4**, 131 (2019).
17. Ulhaq, A., Khan, A., Gomes, D. & Paul, M. Computer Vision For COVID-19 Control: A Survey. *Exp. Proj. with community Collab.* 1–24 (2020) doi:10.31224/osf.io/yt9sx.
18. Voulodimos, A., Doulamis, N., Doulamis, A. & Protopapadakis, E. Deep Learning for Computer Vision: A Brief Review. *Comput. Intell. Neurosci.* **2018**, 1–13 (2018).
19. Iqbal Firdaus, M., Innar Ridho, I., Kalimantan MAB Jl Adhyaksa No, I. & Tangi Banjarmasin, K. APLIKASI PENDETEKSI WAJAH MENGGUNAKAN METODE HAAR (Application of Face Detection Which Using Haar Method). *Media Sains* **9**, 27–31 (2016).
20. Muhathir, M. & Santoso, M. H. Analysis Naïve Bayes In Classifying Fruit by Utilizing Hog Feature Extraction. *J. Informatics Telecommun. Eng.* **4**, 151–160 (2020).
21. Wei, Y., Tian, Q., Guo, J., Huang, W. & Cao, J. Multi-vehicle detection algorithm through combining Harr and HOG features. *Math. Comput. Simul.* **155**, 130–145 (2019).
22. Rahmadani, Lisa; Eriyani, Nadya; Alamsyah, Derry; Devella, S. Implementasi Metode HOG dan Support Vector Machine (SVM) Untuk Multiple Smile Detection. *Stmik Gi Mdp* 1–13 (2019).
23. Tao, P., Sun, Z. & Sun, Z. An Improved Intrusion Detection Algorithm

- Based on GA and SVM. *IEEE Access* **6**, 13624–13631 (2018).
24. Gomes, H., Limao, R. & Santos, A. Feasibility Analysis in Tracking Algorithms for Pattern Recognition through the Lip Articulators. *IEEE Access* **8**, 138907–138921 (2020).
  25. Biswas, D., Su, H., Wang, C. & Stevanovic, A. Speed estimation of multiple moving objects from a moving UAV platform. *ISPRS Int. J. Geo-Information* **8**, (2019).
  26. Alfiani Mahardhika, A., Saptono, R. & Anggrainingsih, R. Sistem Klasifikasi Feedback Pelanggan Dan Rekomendasi Solusi Atas Keluhan Di UPT Puskom UNS Dengan Algoritma Naive Bayes Classifier Dan Cosine Similarity. *J. Teknol. Inf. ITSmart* **4**, 36 (2016).
  27. Saputro, I. W. & Sari, B. W. Naïve Bayes Algorithm Performance Test for Student Study Prediction. *Creat. Inf. Technol. J.* **6**, 1 (2020).
  28. Naiemi, F., Ghods, V. & Khalesi, H. An efficient character recognition method using enhanced HOG for spam image detection. *Soft Comput.* **23**, 11759–11774 (2019).
  29. Antoni, R., Sarwoko, M., Sc, M. & Sunarya, U. Otomatis Dengan Metode Fuzzy Berbasis Mikrokontroller Analysis and Implementation of Sensor System in Automatic Trash. *Anal. Dan Implementasi Sist. Sens. Pada Tempat Sampah Otomatis Dengan Metod. Fuzzy Berbas. Mikrokontroller* **12** (2015).
  30. Mittal, G., Yagnik, K. B., Garg, M. & Krishnan, N. C. Smartphone App to Detect Garbage Using Deep Learning. *Proc. 2016 ACM Int. Jt. Conf. Pervasive Ubiquitous* 940–945 (2016) doi:10.1145/2971648.2971731.

31. Aritonang, P. L. E., Bayu, E. C., K, S. D. & Prasetyo, J. THE PROTOTYPE OF AUTOMATIC SMART TRASH CLUSTERING TOOL. *Snitt* 375–381 (2017).
32. Ariyanto, D., Bahtiar, Y. A., Taufik, M. & Handayani, T. Camera distance effect for smart detection littering Camera Distance Effect For Smart Detection Littering. **030161**, (2020).
33. Wolf, M. *et al.* Machine learning for aquatic plastic litter detection, classification and quantification (APLASTIC-Q). *Environ. Res. Lett.* **15**, (2020).
34. Putra, F. F. & Prabowo, Y. D. Low resource deep learning to detect waste intensity in the river flow. **10**, 2724–2732 (2021).
35. Panwar, H. *et al.* Case Studies in Chemical and Environmental Engineering AquaVision : Automating the detection of waste in water bodies using deep transfer learning. *Case Stud. Chem. Environ. Eng.* **2**, 100026 (2020).
36. Sarker, N., Chaki, S. & Das, A. Illegal Trash Thrower Detection Based on HOG- SVM for a Real-Time Monitoring System. 483–487 (2021).

## DAFTAR LAMPIRAN

### Lampiran 1. Snippet kode program

```

import cv2
import imutils
import numpy as np
import pyautogui
from tracker import *

# Create tracker object
# tracker = EuclideanDistTracker()
tracker = cv2.TrackerCSRT_create()

cap =
cv2.VideoCapture("data/dataset/preprocessing/perbaikan_1.mp4")
# cap = cv2.VideoCapture(0)
# Object detection from Stable camera
object_detector = cv2.createBackgroundSubtractorMOG2(history=100,
varThreshold=40)
# Initializing the HOG person
# detector
hog = cv2.HOGDescriptor()
hog.setSVMDetector(cv2.HOGDescriptor_getDefaultPeopleDetector())
koordinat_awal = 0
arah = 0;
status = "detecting"
def drawBox(img, bbox):
    x,y,w,h = int(bbox[0])-
40, int(bbox[1]), int(bbox[2])+80, int(bbox[3])
    cv2.rectangle(img, (x,y), ((x+w), (y+h)), (255,0,255), 2, 1)
    cv2.circle(img, (int(x+w/2), int(y+h/2)), 10, (255,0,255), 2)
    if koordinat_awal - ((x+w)/2) > 0 :
        arah = 1
    else:
        arah = 0

    pass

while True:
    print(status)
    ret, frame = cap.read()

    frame = imutils.resize(frame, width=640)
    height, width, _ = frame.shape

    print(height)
    print(width)

    # Detecting all the regions
    # in the Image that has a

```



```

w2, y2 + h2), (0, 0, 255), 3)
        cv2.putText(frame, "Littering
Detected", (0, 0 + 20), cv2.FONT_HERSHEY_PLAIN, 2,
                    (0, 0, 255), 2)

cv2.imwrite("data/screenshot/littering_detected.jpg", frame)

    else:
        cv2.putText(frame, "Tracker LOST", (75, 75),
cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)
        (regions, _) = hog.detectMultiScale(frame,
winStride=(8, 8), padding=(4, 4), scale=1.05)
        if len(regions) > 0:
            tracker.init(frame, regions[0])
            status = "tracking"
            koordinat_awal = regions[0][0]
            print("Coordinat awal ", koordinat_awal)
    else:
        (regions, _) = hog.detectMultiScale(frame,
winStride=(8,8), padding=(4, 4), scale=1.05)
        if len(regions) > 0 :
            tracker.init(frame, regions[0])
            status = "tracking"
            koordinat_awal = regions[0][0]
            print("Coordinat awal ", koordinat_awal)

cv2.imshow("mask", mask)
cv2.imshow("Frame", frame)

key = cv2.waitKey(15)
if key == 27:
    break

cap.release()
cv2.destroyAllWindows()

```