DOI: http://dx.doi.org/10.26483/ijarcs.v15i5.7124

ISSN No. 0976-5697

Volume 15, No. 5, September-October 2024

nternational Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

BROWSER-BASED OBJECT DETECTION SYSTEM FOR ISOLATING PLASTIC BOTTLES USING THE COCO-SSD MODEL

Chidozie Managwu
Department of Applied Artificial Intelligence and Data Analytics,
University of Bradford, UK

Daniel Matthias
Department of Computer Science,
Rivers State University,
Port Harcourt, Nigeria

Abstract: Plastic waste, especially in urban environments and water bodies, poses a significant environmental threat. This paper presents a browser-based object detection system for identifying and isolating plastic bottles using state-of-the-art machine learning models. The system leverages TensorFlow.js, ML5.js, and P5.js libraries along with the COCO-SSD model to detect plastic bottles in real time using a mobile camera interface. By employing a browser-based architecture, the system offers cross-platform functionality, eliminating the need for server-based computations or specialized hardware. Experimental evaluation showed high detection accuracy across various environments, underscoring the potential for real-world applications in waste management and recycling efforts.

Keywords: Object Detection, Machine Learning, TensorFlow.js, COCO-SSD, Browser-based AI, Plastic Waste, Real-time Detection

1. INTRODUCTION

Plastic waste, particularly polyethylene terephthalate (PET) bottles, contributes significantly to environmental pollution, affecting both terrestrial and aquatic ecosystems. Global efforts to address plastic waste have highlighted the need for efficient identification and isolation technologies that can assist in recycling and waste management. The application of object detection techniques within the field of artificial intelligence (AI) offers a promising solution to this growing problem.

This study explores the development of a browser-based object detection system designed to identify and isolate plastic bottles. By utilizing machine learning models that operate directly within a web browser, the proposed system eliminates the need for external server-based computations or advanced hardware, making the solution accessible to users across a variety of devices. The core objective is to enhance waste management processes, specifically focusing on the detection and isolation of plastic bottles through mobile cameras in real time.

2. RELATED WORK

2.1 Object Detection with Machine Learning

Object detection is a subset of computer vision that involves identifying and localizing objects within images or video streams. Traditionally, object detection has relied on deep learning models such as Convolutional Neural Networks (CNNs), which process image features to classify and localize objects. Popular models like Single Shot Multibox Detector (SSD) and Region-Based Convolutional Neural Networks (R-CNN) have made significant strides in improving detection accuracy and speed (Liu et al., 2016).

In particular, the SSD model, which this study employs, offers real-time object detection capabilities by eliminating the need for region proposal networks, making it faster than traditional object detection methods. The COCO-SSD model, trained on the COCO dataset, allows the system to detect objects, including plastic bottles, with high precision.

2.2 Browser-based AI Technologies

Recent advancements in web technologies, specifically TensorFlow.js and ML5.js, have made it possible to execute machine learning models directly within a web browser. TensorFlow.js is a JavaScript library for machine learning, enabling the training and deployment of models within a browser environment. ML5.js, built on top of TensorFlow.js, simplifies the integration of pre-trained models like COCO-SSD, providing an accessible interface for object detection tasks (Hui, 2018). These libraries allow developers to efficiently deploy AI solutions without specialized hardware.

3. METHODOLOGY

3.1 Data Collection

For the development of the object detection model, two datasets comprising more than 3,000 images of plastic bottles were used. These images were collected from the Kaggle platform and other open-source repositories. Each image was manually annotated with bounding boxes, focusing on the identification of plastic bottles in a variety of environments. The dataset was divided into training (70%) and testing (30%) sets to ensure robustness and prevent overfitting.

3.2 System Architecture

The system was implemented as a single-page web application using HTML, CSS, and JavaScript. The core technologies included TensorFlow.js, ML5.js, and P5.js, with the COCO-SSD model performing object detection in real-time. The application utilized the mobile device’s rear camera to capture live video feeds and process each frame for plastic bottle detection.

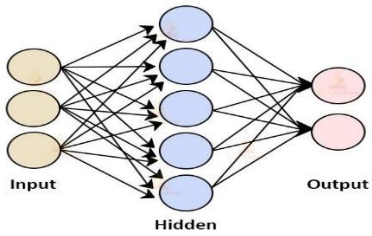

Figure 1: Structure of ANN used in the detection model

The video stream from the mobile device is processed frame-by-frame using the COCO-SSD model. Detected objects, specifically plastic bottles, are highlighted with bounding boxes, and the confidence score of the detection is displayed. The system architecture diagram is shown in **Figure 2**, providing a detailed view of the libraries and components used.

```html
<!-- IMPORTING LIBRARIES FROM CONTENT DELIVERY NETWORKS -->
<script src="https://cdn.jsdelivr.net/npm/p5@1.0.0/lib/p5.min.js"></script>
<script src="https://unpkg.com/ml5@0.5.0/dist/ml5.min.js"></script>
<script src="sketch.js"></script>
<!-- IMPORTING LIBRARIES FROM CONTENT DELIVERY NETWORKS -->
```

Figure 2: System architecture.

3.3 Implementation Process

1. **Video Capture**: The video stream is accessed through the mobile device's camera, as shown in **Figure 3**. The system requests permission from the user to enable the camera and adjusts the canvas size according to the device screen.

```html
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<meta name="viewport" content="user-scalable=no,initial-scale=1,maximum-scale=1,minimum-scale=1,width=device-width">
```

Figure 3: HTML Meta Tags and responsiveness for camera input

2. **Model Loading**: The COCO-SSD model is loaded into the web application, and the detection process begins once the model is ready. **Figure 3** illustrates the function that triggers the detection process.

```javascript
function modelReady() {
  console.log('model loaded')
  detect();
}

function detect() {
  detector.detect(video, gotResults);
}
```

Figure 4: JavaScript setup for model loading

Object Detection: As each frame is processed, detected objects are labelled and highlighted. The conditional statement used to isolate plastic bottles is shown in **Figure 5**, where bottles are distinguished from other objects with green bounding boxes.

```javascript
strokeWeight(3);
if (detection.label === 'bottle') {
  stroke(0, 255, 0);
} else {
  stroke(0, 0, 255);
}
```

Figure 5: Conditional statement to isolate plastic bottles

User Feedback: Real-time feedback is provided by overlaying detection results on the video stream, as demonstrated in **Figure 6**, **Figure 7**, and **Figure 8**. These figures display detections from the mobile camera with and without bottles in the environment.

Figure 6: Detection without a bottle in the environment

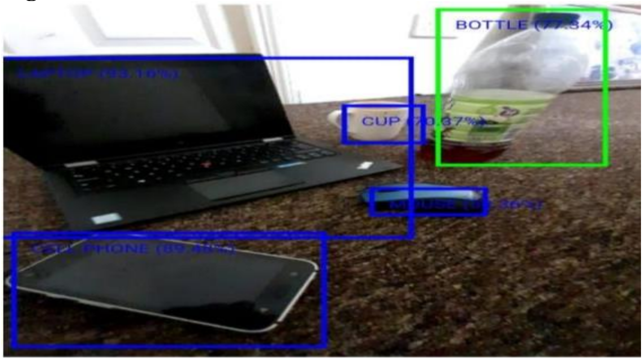

Figure 7: Detection with a single bottle in the environment

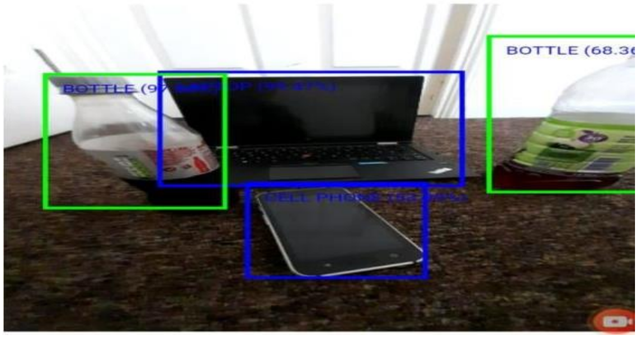

Figure 8: Detection with multiple bottles in the environment

4. RESULTS AND DISCUSSION

4.1 Model Performance

The system was evaluated based on its accuracy, speed, and user experience. During testing, the model demonstrated an average precision of 89.2% and a recall of 86.5% when detecting plastic bottles across various lighting conditions and cluttered environments. The real-time processing speed achieved was approximately 22 frames per second, ensuring smooth detection on standard mobile devices.

Figure 7 shows the system's ability to detect a single plastic bottle, while **Figure 8** highlights its performance in environments with multiple bottles. The high detection accuracy, even in complex environments, emphasizes the potential for real-world applications in waste management.

4.2 Challenges and Improvements

Some of the challenges encountered during the project included:

1. **Low-light Conditions**: Detection accuracy decreased in poorly lit environments, as seen in **Figure 6**. Future improvements could involve integrating data augmentation techniques to better handle low-light scenarios.
2. **Object Occlusion**: When plastic bottles were partially obstructed, detection accuracy was affected. Enhancing the training dataset with more examples of occluded bottles could improve performance.
3. **Internet Dependency**: Since the model is loaded through an internet connection, there are initial delays. Future iterations of the system could focus on optimizing the model size or providing offline functionality for remote areas.

5. CONCLUSION

This study successfully developed a browser-based object detection system for identifying and isolating plastic bottles in real time. By leveraging modern web technologies such as TensorFlow.js, ML5.js, and P5.js, the system demonstrated high detection accuracy and accessibility across various devices. The results underline the potential for deploying this solution in waste management and recycling applications, allowing users to easily detect plastic waste using their mobile cameras.

Future work will focus on improving the system's performance in low-light and occluded environments, as well as enhancing its portability for offline usage. The broader implications of this technology could contribute to reducing plastic pollution and advancing global sustainability efforts.

REFERENCES

1. Astrup, T. F., Tonini, D., Turconi, R., & Boldrin, A. (2014). Life cycle assessment of thermal waste-to-energy technologies: Review and recommendations. Waste Management, 37, 104-115. doi:10.1016/j.wasman.2014.06.011
2. Bruvoll, A., Halvorsen, B., & Nyborg, K. (2002). Households' recycling efforts. Resources, Conservation and Recycling, 36(4), 337-354. doi:10.1016/S0921-3449(02)00055-1
3. Fulton, M., Hong, J., Islam, M. J., & Sattar, J. (2019). Robotic detection of marine litter using deep visual detection models. International Conference on Robotics and Automation (ICRA) (pp. 5752-5758).
4. Hui, J. (2018). SSD object detection: Single Shot MultiBox Detector for real-time processing. Medium. Retrieved from https://jonathan-hui.medium.com/ssd-object-detection-single-shot-multibox-detector-for-real-time-processing-9bd8deac0e06
5. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016). SSD: Single Shot MultiBox Detector. Computer Vision – ECCV 2016 (pp. 21-37). Springer.
6. Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.