International Journal of Engineering Trends and Technology
ISSN: 2231–5381 / https://doi.org/10.14445/22315381/IJETT-V73I4P123

Volume 73 Issue 4, 258-278, April 2025

Original Article

HuYolo-NAS: Real-time Sorting of Recyclable Solid Waste With An Adaptive Neural Network

Bruno Muchotrigo-Albertis, Wilder Reyes-Huanca, Guillermo Zarate-Segura, Luis Hermoza-Paz

Facultad de Ingeniería y de Sistemas, Universidad Tecnológica del Perú, Lima, Perú.

Corresponding Author : u20224252@utp.edu.pe

Received: 18 November 2024 Revised: 20 March 2025 Accepted: 29 March 2025 Published: 26 April 2025

Abstract - *This research addresses the challenges posed by the high environmental variability in waste disposal sites and the inherent inaccuracies in manual waste classification by introducing HuYOLO-NAS, an adaptive neural network model designed to enhance the precision of real-time classification of recyclable solid waste. The system integrates the YOLO-NAS architecture with the Hu moments algorithm to optimize object detection and spatial localization. The model was trained on the 'EcoSight' dataset, comprising 8,400 annotated images of paper, cardboard, PET plastic and hard plastic. Performance was quantitatively assessed using metrics such as precision, recall, F1 score, accuracy, and mean Average Precision (mAP), supplemented by confusion matrix analysis. The results underscore HuYOLO-NAS's potential as an advanced solution for automated waste sorting, reducing the manual labor involved and mitigating sanitation risks, thus providing a robust foundation for future advancements in machine vision for waste management.*

Keywords - *Solid waste management, Adaptative Neural Network, Hu moments, Computer Vision.*

1. Introduction

The implementation of the Comprehensive Solid Waste Management Law, enacted by the Ministry of Environment in 2016, has been hampered by the incompetence of regional and municipal governments in Peru [1]. This law aims to manage and reuse recyclable materials in accordance with environmental and economic principles. However, the lack of adequate infrastructure, poor inter-institutional coordination and the absence of an environmental awareness plan have resulted in ineffective management of solid waste in the country [2]. In Peru, the recycling rate is 1.9% [3], much lower than the regional average of 4.6% in Latin America and the Caribbean; for example, Colombia has a recycling rate of 10.6% [4]. This low rate generates an excessive accumulation of waste in landfills and public spaces [5], which exposes personnel in charge of manual solid waste separation to health risks and hinders the recovery of recyclable material [7]. It also contributes approximately 5% to greenhouse gas emissions [6]. It is therefore necessary to address solid waste management to improve public health and reduce its environmental impact. In the districts of the provinces of Lima and Callao, in 2020, less than 1% of the total municipal solid waste generated will be recovered. This value is well below the average of developed countries, where approximately 35% of water is recycled or composted. This low recovery rate reflects deficiencies in the "implementation of an integrated municipal solid waste management system" [8]. Different investigations have addressed the effective management of solid waste; for example, a smart garbage collection model was implemented in Norway that uses trucks to collect bags of different colors, corresponding to different types of waste, which are then processed in an optical sorting plant. This system, implemented in 2012, has a precision of 98% and managed to increase the recycling rate by 37% by 2018 [9]. Additionally, in Australia, machines with artificial vision have been implemented to process up to 160 recyclables per minute, compared to the average of 35 recyclables per minute achieved by human labor [10]. In another case study from India, a computer vision system based on convolutional neural networks was developed to classify electronic waste, classifying waste into eight categories with an accuracy of 96%. It is projected that the execution of this project could reduce costs by up to 20% by 2026 if manual sorting is replaced [11]. In this way, the artificial vision has proven its usefulness by increasing the recycling rate [8], reducing human labor [10], and decreasing the costs of sorting recyclable solid waste [11].

However, there is a general research gap in machine vision application in environments characterised by high variability, as is the case in the Peruvian environment. In such environments, the diversity and heterogeneity of waste in landfills and disposal sites represent an additional challenge for adopting automated sorting systems. This issue highlights the need to develop adaptive solutions that respond to the specific conditions of highly variable environments.

Consequently, the present research aims to develop and implement HuYOLO-NAS, an adaptive neural network model to integrate the YOLO-NAS architecture with the Hu moments algorithm to classify recyclable solid waste in real-time. This innovative approach is adapted to the particularities of the Peruvian environment, offering a more accurate and efficient solution to optimise the separation process in recycling, in contrast to conventional methods.

2. State of the Art

2.1. Classification of Solid Waste

Solid waste is defined as materials dumped into ecosystems in a semi-solid or solid state and managed through collection, treatment, marketing and reuse processes. They are also classified as plastic, paper and cardboard [12]. They are also categorized according to their origin and reuse potential: recyclables include plastics, glass, metals and textiles, while non-recyclables include organic matter and hazardous materials [13].

2.2. Machine Vision Approaches in Solid Waste Classification

Machine vision is a discipline of Artificial Intelligence (AI) that emulates the functionality of the sense of sight, allowing the collection, processing and analysis of spatial data from digital images [14]. Neural networks consist of interconnected nodes that learn from data for recognition, classification and prediction tasks [15]. The main architectures are Recurrent Neural Networks (RNN), suitable for time series and natural language, and Convolutional Neural Networks (CNN), which process spatial features in images to identify visual patterns [16]. Likewise, learning algorithms are classified into supervised ones for labeled data, unsupervised ones for detecting patterns in unlabeled data and reinforcement learning algorithms that learn through rewards and penalties [17].

Different approaches have been explored for solid waste classification using computer vision. On the one hand, deep learning-based neural networks have been used in the DeepSORT You Only Look Once - trash (DSYOLO-trash) algorithm [18] and the Garbage Classifier Deep Neural Network (GCDN-Net) [19]. Both are known for their high performance in automated capturing and extracting abstract features in high variability scenarios. On the other hand, the ShuffleNet v2 algorithm powered with Yolov5s [19] used the architecture of convolutional neural networks; this approach differs from DSYOLO-trash [18] and GCDN-Net [18] by being efficient on computers with limited data processing capacity. Similarly, the k-nearest-neighbors supervised learning algorithm was used; this method is versatile with input data and is fed back with output data, optimizing the object detection algorithm iteratively and autonomously [11]. Thus, each type of algorithm has a specific application in computer vision systems, which directly influences the results desired in the effective management of solid waste.

2.3. Datasets

Image databases are crucial for training accurate algorithms for solid waste recognition. Two datasets were used: “TrashNet”, with 2528 images of glass, paper, cardboard, plastic, metal and trash, and “MMTrash”, with 2332 multi-labeled images of mixed waste. These datasets allowed for obtaining a recognition accuracy of 98.5% [18]. In addition, the “Garbage In, Garbage Out” dataset was used, with 25,000 processed and labeled images, reaching an accuracy of 95.77% [19].

Additionally, a dataset of 6632 images was created using a web crawler and mobile captures. Applying enhancement techniques, such as random rotations and brightness adjustments with the Python imgaug library, generated a total of 7072 labeled frames in ten categories. Finally, the data were distributed in a ratio of 80:20 for the training sets and the validation and test sets, achieving an accuracy of 94% [20]. This shows that careful data selection and preprocessing can significantly improve the accuracy of solid waste recognition.

2.4. Efficiency of Classification Algorithms in Variable and Controlled Environments

During the transportation of solid waste on a moving conveyor belt, a notable difference in the performance of object tracking algorithms is observed depending on the type of environment. On the one hand, the DSYOLO-Trash model [18] has shown significant improvements in accuracy when identifying solid waste in real-time in highly variable environments, defined as those with operating conditions that change frequently and unpredictably, where stability is difficult to maintain, and continuous adaptation is essential [16]. On the other hand, the Oriented Fast and Rotated BRIEF (ORB) algorithm showed greater accuracy in classification in a first-class controlled environment characterized by constant illumination, homogeneous backgrounds and calibrated cameras. These optimized conditions allow ORB to outperform Speeded Up Robust Feature (SURF) and Scale Invariant Feature Transform (SIFT) [21]. This comparison shows how the effectiveness of the algorithms varies depending on the stability of the environment in which they are applied.

2.5. Robotic Arm Control

Robotic arms improve productivity in processes such as waste management by handling hazardous objects but face challenges in accuracy, strength, and speed. The adaptive control model with dual neural networks (ADNSMC) outperforms other methods in accuracy and robustness, compensating for kinematic and dynamic uncertainties with recurrent neural networks (RNN) and radial basis fencing networks (RBFNN) [20]. Furthermore, an approach to calculate the position and accuracy of industrial robots such as KUKA RSI reduced orthogonal deviations by more than 64% and inline errors by more than 93%, greatly improving accuracy in linear movements at 55 mm/s [21]. In that sense,

the precision of the robotic arm would be guaranteed to move recyclable waste to further treatment.

2.6. Adaptive Neural Networks in Computer Vision

Adaptive neural networks dynamically adjust their structures and parameters during inference in relation to the input data. These networks optimize resources by selectively activating model components on demand and support advanced optimization techniques such as quantization, pruning, and knowledge distillation. As a result, they improve efficiency, representation power, and adaptability to different computational budgets. In comparison, static models such as RNNs and CNNs maintain a fixed computational graph and parameters after training [22].

Thus, implementing an adaptive neural network can update the computer vision system in real time and improve the performance and detection of solid waste. The purpose of the research is to classify recyclable solid waste using computer vision to optimize separation in recycling in highly variable environments. In this sense, the following steps have been proposed to fulfil the purpose. First, a dataset with four types of solid waste that can be recycled will be created.

Secondly, the HuYOLO-NAS model will be trained for the accurate classification of solid waste. Thirdly, a robotic arm will be implemented to transfer the solid waste to its corresponding containers based on the information provided by the artificial vision model. Finally, the classification accuracy of the system will be evaluated to measure its efficiency.

3. Methodology

3.1. Creating the Dataset

The authors of this research study created the EcoSight dataset, which is available in a public GitHub repository, HuYolo-NAS [23]. The dataset comprises 8400 RGB images in Joint Photographic Expert Group (JPG) format with a resolution of 640 x 640 pixels. Additionally, the images are organized into four categories, focusing on Peru's most recovered and valued recyclable solid waste: PET plastic, hard plastic, paper and cardboard [24].

The PET plastic category included images of containers used for water, rehydrating drinks, liquid dishwashing liquid, rinse aids, disinfectants, oil and vinegar. Regarding hard plastic, images of yogurt containers, lids of various beverage containers, bleach containers, disposable white plastic cutlery, solid dishwashing liquid and butter containers will be collected. The paper category will include images of mixed paper, white paper, and coated paper, as well as manila envelopes, folders, glossy paper, and ballots. Finally, the Cardboard category will refer to images of cardboard pieces and cardboard boxes. The stages of creating the EcoSight dataset (Figure 1) comprised 4 stages: image collection, preprocessing, labelling and enlargement.

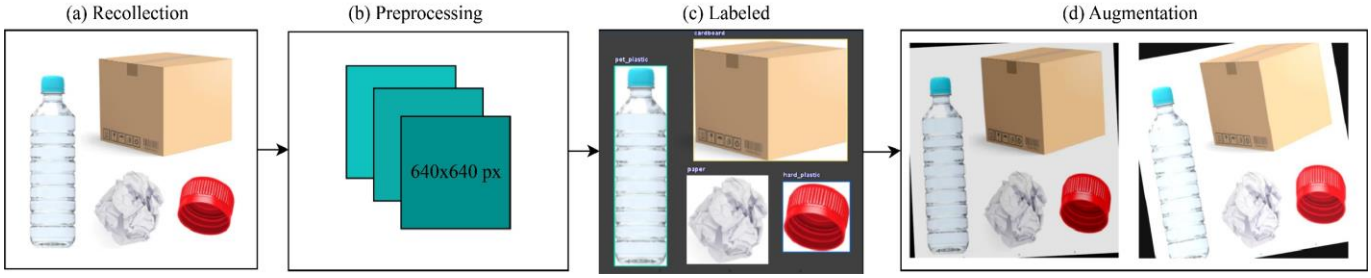

Fig. 1 Stages of EcoSight dataset creation: (a) Recollection; (b) Preprocessing; (c) Labelling; (d) Augmentation

3.1.1. Image Collection

The image collection used was of a mixed nature, integrating both automated sources and our own collection, which allowed us to guarantee consistent diversity and quality in the dataset. The distribution of the images is presented in Table 1 by the type of collection used and the type of class. Firstly, the automated collection was carried out through the Roboflow Universe platform, with 1600 images obtained and distributed in 250 PET plastic images, 320 hard plastic images, 494 paper images and 536 cardboard images. In this way, the Roboflow Universe platform facilitated this process under the AGPL-3.0 open-source license, allowing access to high-quality images with a focus on transparency and reuse of the images for research purposes. Secondly, a private collection was carried out in recycling centres in Metropolitan Lima, where 1600 images were captured and distributed in 250 PET plastic images, 319 hard plastic images, 495 paper images and 536 cardboard images. This approach involved taking the images on a white background to ensure visual consistency and minimize external interferences for optimal data quality control.

Table 1. Distribution of images by class and type of collection

Class Name	Automated collection	Private compilation	Total
PET Plastic	250	250	500
Hard Plastic	320	319	639
Paper	494	495	989
Cardboard	536	536	1072
Total	1600	1600	3200

Note: The table shows the total number of images per class, differentiated by automated and private collection.

3.1.2. Image Preprocessing

For automated image preprocessing, a Python script with the PIL library was used, where the variables source_folder and destination_folder define the paths of the source and destination folders for the four recyclable solid waste categories. The script resizes each image to 640x640 pixels and creates the destination folders automatically with os.makedirs(destination_folder), ensuring an organized structure. Then, it iterates through each image in source_folder, processes it with img.resize, and saves it to the location specified in destination_folder. In this way, we follow the recommendations of Fitzgerald et al. [28], who demonstrated that adequate image resolution significantly improves the accuracy of models in computer vision tasks. This automated preprocessing process ensures that the 3200 collected images conform to size standards before moving on to the labeling process, thus optimizing the preparation of the dataset for use.

3.1.3. Image Labelling

The adaptive neural network requires labeled data, so bounding boxes were used in the 3200 collected images. In this sense, the Manual Labeling technique was used, where the authors labeled the images with bounding boxes. Additionally, the annotation classes corresponding to the categories: “pet_plastic”, “hard_plastic”, “paper” and “cardboard” were configured, and the annotation tool classes were blocked to ensure that only these defined categories were used. In addition, quality control was implemented, and every 200 labeled images were reviewed by one of the authors to detect and correct possible errors.

After completing the bounding box labeling, an adequate balance between classes was achieved. In total, approximately 5764 annotations were made. The annotated classes were as follows: cardboard with 1409 annotations, hard plastic with 1181 annotations, paper with 1111 annotations, and PET plastic with 2063 annotations. The distribution of annotations by class is shown in Table 2.

The technical information of the EcoSight dataset (Table 3) shows that the average image size was 0.41 MP, spanning a range from 0.05 MP to 12.19 MP, and the average resolution was 640x640, maintaining a square aspect ratio. In addition, no missing annotations or null examples were found, ensuring data consistency for the machine vision system for recyclable solid waste classification.

3.1.4. Data Expansion

Lin et al. [29] claim that image augmentation techniques should be employed to improve model performance. In this context, the existing 3200 solid waste images dataset will be manipulated using specific techniques, including 90° rotations (clockwise, counterclockwise and reverse) and additional rotations between -15° and +15°. Furthermore, saturation levels will be adjusted between -25% and +25%, and brightness between -15% and +15%, in addition to applying a blur of up to 2.5 px. These techniques were executed in Roboflow and are intended to reduce overfitting while increasing the number of training images. Finally, 8400 images were obtained, which were divided into 93% for training with 7800 images, 2% for validation with 200 images and 5% for testing with 400 images.

Fig. 2 Images from the EcoSight dataset: (a) PET Plastic (b) Hard Plastic (c) Paper (d) Cardboard

Table 2. Distribution of annotations by class

Class Name	Automated collection
pet_plastic	2063
hard_plastic	1181
paper	1111
cardboard	1409

Note: The table shows the distribution of annotations made for each class of recyclable solid waste in the labelling process.

Table 3. Technical information of the dataset

Characteristic	Value
Imagens	3200
Annotations	5764
Classes	4
Average size per image	0.41 MP
Image size range	0.05 MP - 12.19 MP
Medium resolution	640x640 px

Note: The table reflects the technical characteristics of the EcoSight dataset.

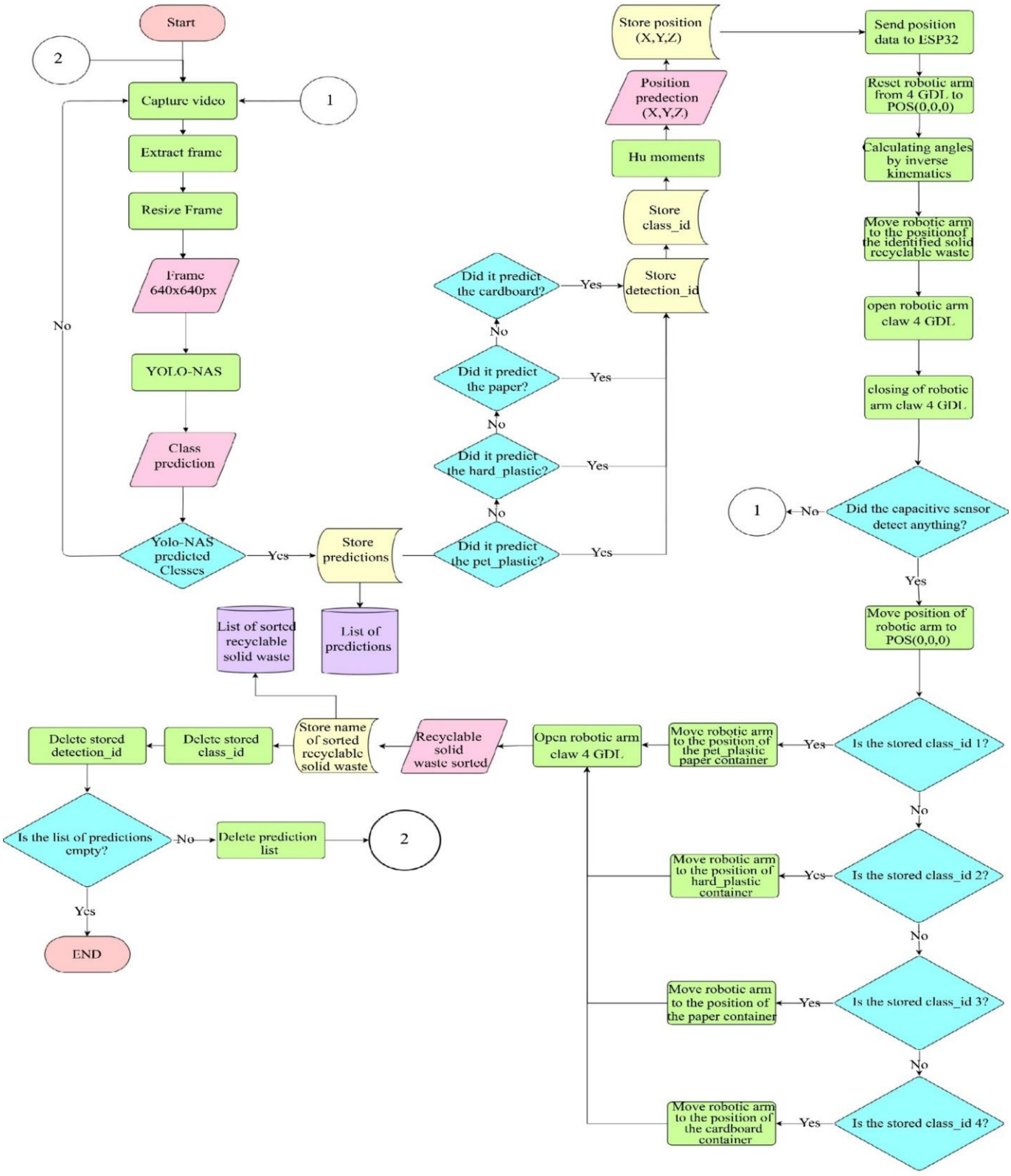

Fig. 3 Flow chart of detection and classification of recyclable waste with HuYOLO-NAS

3.2. HuYOLO-NAS Model

The HuYOLO-NAS model was developed, which integrates the YOLO-NAS computer vision model [30] and the Hu moments algorithm [31] and will be used to classify and calculate the real-time position of four classes of recyclable solid waste. The process (Figure 3) begins with real-time video capture, from which the individual frames are extracted. These frames are then resized to 640 x 640 pixels. The YOLO-NAS neural network is then used to predict the objects' classes in the extracted frames. In case no class is detected, the process concludes at that point and returns to the real-time video capture. However, if any waste class is detected, the predictions are stored in a list for further processing. Depending on the predicted class of waste (pet_plastic, hard_plastic, paper or cardboard) and following an established priority order for detection, its corresponding class_id and prediction_id are stored in the list. Following the identification of the classes, the three-dimensional position of the object, X, Y, and Z coordinates, is calculated from the Hu moments algorithm to determine its exact location in space. These data, both class and position, are transmitted to the ESP32 microcontroller, which is responsible for managing the movement of the robotic arm. The displacement of the arm is calculated using inverse kinematics algorithms to ensure optimal accuracy in locating the object. Upon reaching the predicted position, the claw of the robotic arm opens to grab the waste. Once the capacitive sensor confirms the detection of the object, the claw closes to proceed with its collection. Subsequently, depending on the predicted waste class (such as PET plastic, hard plastic, paper or cardboard), the robotic arm moves to the appropriate container to deposit the object. This classification process implies that the system stores both the name of the waste and its class, ensuring an accurate record of the processed waste. Once the object has been deposited, the corresponding predictions are removed from the pending task list. Finally, the prediction list is checked to see if it is empty. If it is, the process stops; otherwise, the cycle is repeated to continue sorting the remaining waste. In this way, the procedure efficiently automates the detection, sorting and disposal of recyclable solid waste using an artificial vision system combined with a robotic arm.

3.2.1. The YOLO-NAS Model

Recyclable solid waste classification is performed using You Only Look Once - Neural Architecture Search (YOLO-NAS), which implements an advanced object detection system optimized through a four-phase architectural search [32]. First, in the "Input" phase, images are received. Then, the "Backbone" phase processes them through a CNN that extracts edges and textures. Next, the "Neck" phase refines these features using feature pyramids. Finally, the "Head" phase predicts the class of each detected recyclable solid waste. The performance of YOLO-NAS was optimized using the Neural Architecture Search technology [33], which adjusts the model architecture to balance accuracy and efficiency. In addition, the model applies quantization through the Hybrid Quantization method, where the accuracy is reduced in specific parts of the model using Quantization-Aware Blocks [34] and Selective Quantization [35]. In addition, INT8 quantization is used to reduce memory usage and inference times, making it suitable for edge devices with limited hardware, such as the Tesla T4 GPU used for model training. Finally, its performance is enhanced by the SuperGradients Training Toolkit library, which provides an optimized environment for model training and tuning, allowing for greater efficiency and accuracy in detection. Therefore, YOLO-NAS is a robust option for real-time classification of four classes of recyclable solid waste [37].

3.2.2. The Hu Moments Algorithm

To implement the Hu Moments algorithm, OpenCV was used for processing captured images in real time, converting them to grayscale, and binarizing them using thresholding. These operations are necessary to prepare the image before calculating the Hu Moments. NumPy was used to manipulate the numerical data and facilitate matrix operations and bounding box coordinates. In addition, supervision was used to manage detections and bounding boxes, allowing the identification of objects and their classes.

Table 4. HuYOLO-NAS training hyperparameters

Parameter	Value
warmup_initial_lr	1e-6
lr_warmup_epochs	3
initial_lr	2e-4
cosine_final_lr_ratio	0.3
weight_decay	0.001
decay	0.9
max_epochs	20
score_threshold	0.1
top_k_predictions	300
num_classes	4
nms_top_k	300
max_predictions	100
nms_threshold	0.7

Note: The table shows the hyperparameters used in training.

The Hu Moments are a set of seven descriptors calculated from the normalized moments of the object, which are invariant under geometric transformations such as rotation, translation and scale. These descriptors are defined by the following equations:

$$
\begin{aligned}
hu[0] &= n_{02} + n_{20} \\
hu[1] &= (n_{02} - n_{20})^2 + 4n_{11}^2 \\
hu[1] &= (n_{02} - n_{20})^2 + 4n_{11}^2 \\
hu[2] &= (n_{30} - 3n_{12})^2 + (3n_{21} - n_{03})^2 \\
hu[3] &= (n_{30} + n_{12})^2 + (n_{21} + n_{03})^2 \\
hu[4] &= (n_{30} - 3n_{12})(n_{30} + n_{12})[(n_{30} + n_{12})^2 \\
&\quad - 3(n_{21} + n_{03})^2] + (n_{21} - 3n_{03})(n_{03} \\
&\quad + n_{21})[3(n_{30} + n_{12})^2 - (n_{21} + n_{03})^2]
\end{aligned}
$$

$$hu[5] = (n_{20} - n_{02})[(n_{30} + n_{12})^2 - (n_{21} - n_{03})^2] + 4n_{11}(n_{30} + n_{12})(n_{21} + n_{03})$$
$$hu[6] = (3n_{21} - n_{03})(n_{21} + n_{03})[3(n_{30} + n_{12})^2 - (n_{21} + n_{03})^2] + (n_{30} - 3n_{12})(n_{21} + n_{03})[3(n_{30} + n_{12})^2 - (n_{21} + n_{03})^2] \quad (1)$$

Once the object is detected, the corresponding bounding box is used to crop the image's region of interest (ROI). This process involves extracting the coordinates of the bounding box and cropping the image based on those coordinates. The region of interest is the area of the image that contains the detected object. The cropped image is then converted to grayscale to simplify the analysis, and a binary threshold is applied to separate the object from the background. Binary thresholding converts the image to black and white, where the object is rendered in white (value 255) and the background in black (value 0). This highlights the shape of the object and removes irrelevant details. Next, geometric moments of the binarized image are calculated using OpenCV's cv2.moment's function. These moments are statistical properties of the binarized image that describe the distribution of pixels. These moments are then transformed into Hu Moments using the cv2.HuMoments function. Finally, the Hu Moments are used to calculate the three-dimensional coordinates of the object, which are derived from the first three moments. These coordinates are sent to the ESP32 microcontroller to perform the manipulation and classification actions on the detected object.

3.3. Training of the HuYOLO-NAS Model

First, the runtime environment in Google Colab is configured to use the T4 GPU. Then, the super-gradients, albumentations, imutils, roboflow, pytube, supervision, and onemetric packages are installed. From the super_gradients package, Trainer is included to manage the training of the HuYolo-NAS model, dataloaders for data loading, as well as the coco_detection_yolo_format_val and coco_detection_yolo_format_train functions from dataloaders, which allow the use of data in YOLO format for training and validation. These imports facilitate detection and classification during the training process. Additionally, PPYoloELoss is incorporated for the PP-YOLO loss function, DetectionMetrics_050 for the calculation of detection metrics, and PPYoloEPostPredictionCallback for the post-processing of predictions. Also, the model's module, which manages the detection models, is included. Next, the directory for the checkpoints and the name of the experiment are set. The variable CHECKPOINT_DIR is defined as 'checkpoints2', specifying the location where the model's checkpoints will be saved. Then, a training instance is created using the Trainer class, assigning the experiment's name as 'sign_yolonas_run2' and the checkpoint directory as CHECKPOINT_DIR, which allows the training results to be organized and stored efficiently. The dataset is exported from Roboflow to the Google Colab environment, and then the dataset parameters are loaded into a dictionary, which facilitates access to the information during training. In this dictionary, the path to the main data directory is defined, specifying data_dir, as well as the names of the directories for the training, validation, and test set images and labels. Finally, a list of class names is included under the classes key, allowing for structured data management during the training process.

Then, the values from the dataset_params dictionary are assigned to the corresponding arguments to create the training, validation, and test datasets. For the training set, the coco_detection_yolo_format_train function defines the dataset parameters, including the path to the data directory, the training image and label directories, and the classes. The dataloader parameters are also configured, setting a batch_size of 16 and num_workers of 2. Similarly, the validation dataset is configured using coco_detection_yolo_format_val, specifying the validation image and label parameters. In addition, the test dataset is created with the same function, using the appropriate parameters for the test images and labels. Next, the model is created, now configured for finetuning. The yolo_nas_s model is instantiated using models.get(), specifying the number of classes with num_classes=len(dataset_params['classes']) and loading the pre-trained weights from coco.

To define the training hyperparameters (Table 4), the Adam optimizer has been chosen, and additional parameters have been tuned, such as optimizer_params, which includes a weight_decay of 0.001, and EMA decay has been enabled (ema_params) with a value of 0.9. The initial learning rate (initial_lr) is set to 2e-4, with a cosine mode schedule and an initial warmup of lr_warmup_epochs for three epochs. Furthermore, training is performed with mixed precision (mixed_precision) and runs for 20 epochs. The loss function chosen is PPYoloELoss, set to the number of classes in the dataset, while DetectionMetrics_050 in valid_metrics_list will monitor the accuracy with a threshold of 0.1. To start training the model, the trainer.train command is run, in which the model to be trained (model), the training parameters (train_params), the training dataset (train_data), and the validation dataset (val_data) are specified. This process allows the model to be tuned according to the predefined metrics and configurations to optimize its performance.

Finally, to obtain the best-trained model, the models.get function is used, specifying the name of the model, in this case, yolo_nas_s, the number of classes that are determined based on the dataset parameters, and the path to the checkpoint file (checkpoint_path) located at checkpoints2/sign_yolonas_run2/ckpt_best.pth. This file stores the most optimized version of the model, including the weights and configurations resulting from the training process, ensuring that the best results obtained are used, thus improving accuracy and performance in the task of detecting the four classes of recyclable solid waste: paper, cardboard, pet plastic, and hard plastic.

3.4. Design of the Robotic Arm "HuArm"

3.4.1. Description of the Robotic Arm

The robotic arm developed in this project was a mechatronic system with 4 Degrees of Freedom (DOF) intended to simulate an automated handling system for sorting recyclable solid waste. This model was structured based on a modular design of an industrial manipulator adapted for a high variability environment, such as solid waste sorting [38].The design considered the implementation of articulated links, and each mounted on a servomotor that provided sufficient torque to support and manipulate the links at each joint of the arm, maintaining precision and stability in the movements. Also, a gripping claw was included, and a design based on a finger gripper was chosen, controlled by a servomotor, which allowed precise manipulation of the objects during the tests.

The control system (Figure 4) of HuArm integrates a power source that provides electricity to the entire system, thus ensuring the continuous operation of each module. Next, the HuYolo-NAS algorithm calculates the object's position and transmits this information to the ESP32 controller. In parallel, a capacitive sensor determines whether there is contact with the recyclable solid waste, sending this signal to the controller as well. With this data, the controller performs calculations to determine the optimal speed and angle of movement using inverse kinematics, which are then sent to the servomotors. Finally, the servomotors execute the instructions received, generating the precise movement that allows the robotic arm to place the solid waste in its respective container.

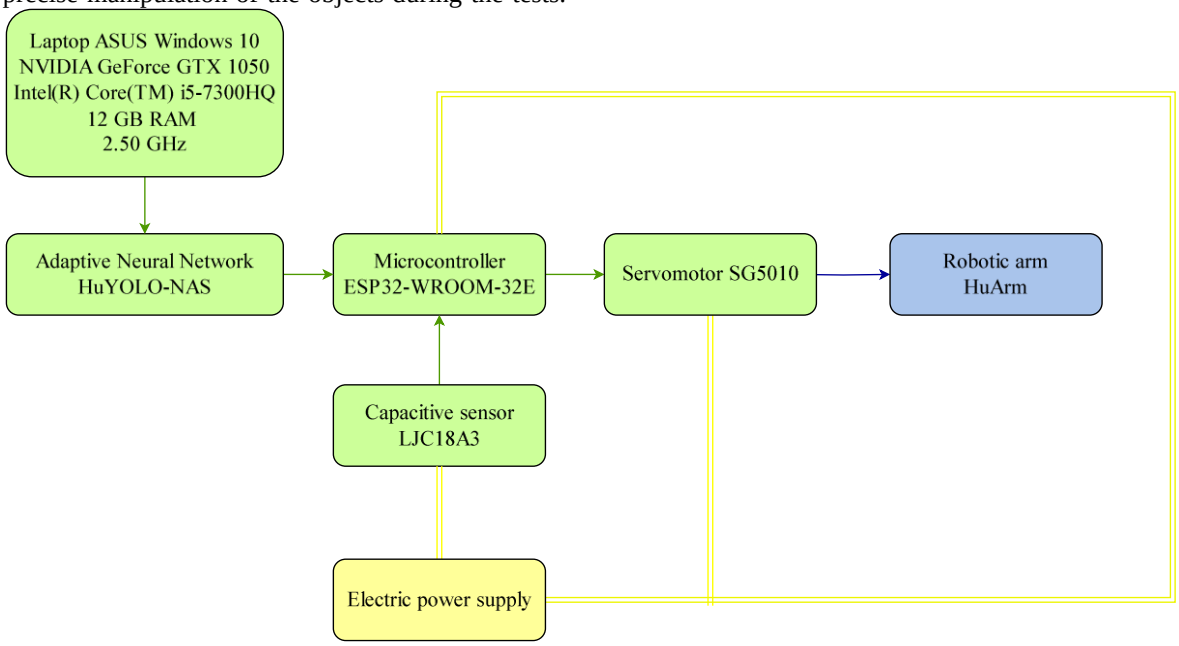

Fig. 4 Block diagram of HuArm operation

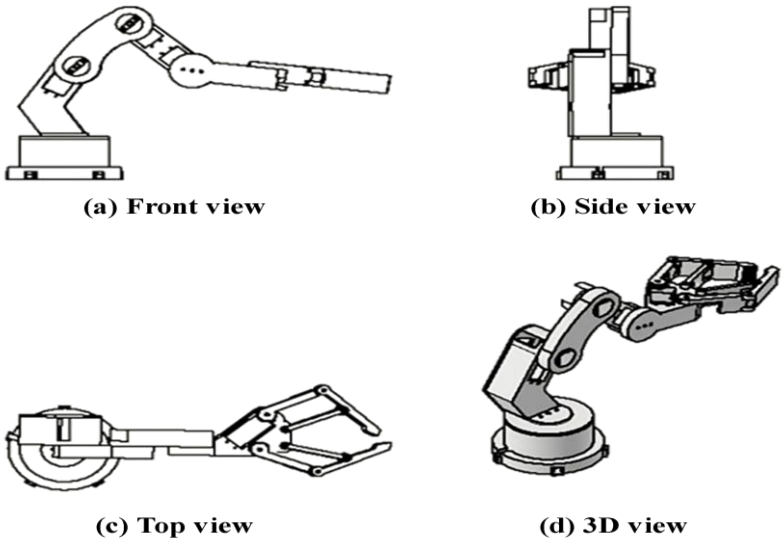

Fig. 5 HuArm design in Autodesk Inventor: (a) Front view, (b) Side view, (c) Top view, (d) 3D view

Table 5. Table of results of strength and torque in the joints

Joint	Force (N)	Torque (N·m)	Torque (kg·cm)
Gripper	5.14	0.46	4.72
Joint 3	0.24	0.48	4.93
Joint 2	0.24	0.51	5.15
Joint 1	0.24	0.53	5.37
Base	0.64	0.58	5.95

Note: Force and torque values have been calculated for each joint, considering mass and distance from the rotation point.

Table 6. Technical information on the SG5010 servo motor

Features	Details
Dimensions	40.2*20.2*43.2 mm
Torque	3.1Kg-cm (4.8V); 6.5Kg-cm (6.0V)
Operating Speed	0.17seg/60º (4.8V no load); 0.4seg/60º (6V)
Operating Voltage	4.8-6 Volts

Note: The table reflects the technical characteristics of the SG5010 servo motor provided by its datasheet from the company Tower Pro Pte Ltd.

Table 7. Technical information on the LJC18A3 capacitive sensor

Features	Details
Dimensions	D18mm*L70mm
Operation Voltage	6-36V DC
Working Current	20mA max.
Output Current (load)	300mA max.
Detection range	1mm to 5mm
Out	NPN normally open type
Weight	108 gr.

Note: The table reflects the technical characteristics of the LJC18A3 capacitive sensor provided by its datasheet from the company YUEQUING HENGWEI ELECTRONICS CO., LTD.

3.4.2. Dimensions of the Robotic Arm

For the realization of the robotic arm (figure 4), 4 DOF were considered because this type of model provided the ability to position and orient the tool precisely at any point in the work area [39]. The sizing was done in the Autodesk Inventor design environment using tools such as Line, Circle, Rectangle, Circular, Fillet, and Extrusion, among others. During the modelling, the base was started and progressively advanced towards the joints until finishing with the sizing of the claw intended for holding the solid waste. The final model, to scale, was printed in 3D using PLA material, which allowed for the assurance of structural precision.

3.4.3. Selection of Servomotors

For the selection of the servomotors, the calculation of the torques required for each joint was performed using Craig's formula. Therefore, the dot product between the vector describing the direction of the joint axis and the vector of moments acting on the link connected to said joint had to be calculated [39].

$$\tau_i = \mathrm{n}_i^T \cdot \mathrm{Z}_i \quad (2)$$

τ_i: It is the torque required at joint i.

n_i: It is the vector that describes the direction of the axis of joint i.

Z_i: It is the vector of moments acting on the link connected to joint i.

In a robotic arm, since the force and the lever arm are perpendicular, the formula is simplified by calculating the product of the magnitudes of the force and the distance. Considering the system can manipulate materials such as plastic weighing 500 g, the torque calculations at each joint are adjusted to support both this weight and the arm components. Table 5 presents the force and torque values obtained for each joint, reflecting the effort required at each point of the system.

The results obtained for each robotic arm joint show the force and torque values necessary for its correct operation. In the Gripper, a force of 5.14 N and a torque of 0.46 N m (4.72 kg cm) are calculated, which ensures its handling capacity. As we move towards the joints, such as Joint 3, Joint 2 and Joint 1, the torque values increase slightly, reflecting the additional effort each component must withstand. Finally, in the Base, an increase in force to 0.64 N is observed, with a torque of 0.58 N m (5.95 kg cm), guaranteeing the stability and support of the entire system. With the values obtained from the torques, the selection of the servomotor that met the requirements of a minimum torque of 4.72 kg cm and a maximum of 5.95 kg cm was made. The most suitable servo motor was the SG5010, whose characteristics are presented in Table 6.

3.4.4. Sensor Selection

For selecting the capacitive sensor, criteria such as compact size to be placed on the gripper and low contact sensitivity to avoid errors in detection were considered. With these requirements, the LJC18A3 capacitive proximity sensor was selected, whose characteristics are detailed in Table 7.

3.4.5. Selecting the Controller

The controller selection considered all arm components, including the servo motors and the capacitive sensor. In addition, the memory capacity required to process the data provided by the Machine Vision System was considered. The ESP32-WROOM-32E, the Arduino Nano Every, and the Raspberry Pi Pico (Table 8) are widely used controllers in electronics projects, but they have significant differences in terms of capabilities and features. The ESP32-WROOM-32E is distinguished by its dual-core Tensilica LX6 processor at 240 MHz, which gives it greater processing capacity compared to the Arduino Nano Every, which has an 8-bit processor at 20 MHz. In addition, the ESP32 has 520 KB of RAM, considerably more than the 6 KB of the Arduino Nano Every, allowing it to manage more complex processes and perform simultaneous tasks more efficiently.

Table 8. Comparison of drivers for HuArm

Controller	ESP32-WROOM-32E	Arduino Nano Every	Raspberry Pi Pico
Processor	Dual-core Tensilica LX6 a 240 MHz	ATmega4809 (8-bit, 20 MHz)	Dual-core ARM M.+133 Mhz
RAM	520 KB SRAM	6 KB SRAM	264 KB SRAM
Storage	16 MB Flash	48 KB Flash	2 MB Flash
Connectivity	Wi-Fi y Bluetooth 4.2	Sin Wi-Fi ni Bluetooth	Sin conectividad
GPIOs	34 pines E/S (Configurables como entradas o salidas)	22 pines (14 digitales y 8 analógicos)	26 pines multifunción
Interfaces	I2C, SPI, UART, ADC (12 bits), DAC	I2C, SPI, UART, ADC	SPI, I2C, UART
Operating Voltage	3.3V	5V	3.3V

Note: The table reflects the technical information of each controller selected as an option to control the robotic arm.

In terms of connectivity, the ESP32-WROOM-32E offers native support for Wi-Fi and Bluetooth 4.2, making it ideal for projects that require wireless communication – a feature that the Arduino Nano Every lacks. The Raspberry Pi Pico, meanwhile, also lacks wireless connectivity but offers a high-performance option with its dual-core 133MHz ARM Cortex M0+ processor, paired with 264KB of RAM. However, its lack of wireless connectivity limits its versatility for projects.

Finally, the number of GPIO pins is another relevant factor. The ESP32-WROOM-32E offers 34 configurable pins, making it more suitable for projects that require controlling multiple devices, such as servo motors. The Arduino Nano Every has 22 pins, with a mix of digital and analogue ones, while the Raspberry Pi Pico has 26 multi-functional pins, placing it between the previous two in terms of connection versatility. For the project, the ESP32-WROOM-32E was chosen due to its large number of configurable pins as outputs to control the servo motors and its high processing capacity, thanks to the 520 KB of RAM.

3.4.6. HuArm Overview

The final prototype of the HuArm consists of an ESP32-VROOM-32E controller, 5 SG5010 servomotors, an LJC18A3 sensor and 17 3D-printed parts.

3.5. Experimental Training Environment

In the experimental environment, Google Colab was used for training, using the Tesla T4 GPU. At the end of the process, the file located at checkpoints2/sign_yolonas_run2/ckpt_best.pth was obtained, which allows the model to be run on a computer with the following specifications: an Intel Core i5-7300HQ processor, 12 GB of RAM, Windows 10 Home Single Language operating system (version 22H2), and an NVIDIA GeForce GTX 1050 graphics card. Likewise, Python 3.11.9 and the inference-gpu package in PowerShell were used to integrate inference capabilities.

The CUDA Toolkit 11.8 and cuDNN were used to optimize performance in neural networks. In addition, zlib was used to ensure the operation of the Visual Studio 2019 C++ libraries and runtime to complete the environment configuration, facilitating the execution of the model and the visualization of annotated images. To implement the model, the OpenCV library is imported along with SuperGradients. Through the models.get command, the yolo_nas_s model is loaded configured with 4 classes: 'cardboard', 'hard_plastic', 'paper' and 'pet_plastic', using the previously mentioned weights file. Subsequently, the yolo_nas_s model is converted to ONNX (Open Neural Network Exchange) format, which allows its use in various platforms and machine learning frameworks, facilitating its implementation. The input form specified as (3, 640, 640) indicates that the model accepts 640x640 pixel images with 3 color channels (RGB), which is essential for its correct execution on other systems that support ONNX.

3.6. Evaluation Metrics

To evaluate the performance of the proposed model in classifying recyclable solid waste into four classes, the mean average precision (mAP), precision and recall were used as key metrics. The mAP is a comprehensive indicator that reflects the effectiveness of detection in all categories included in the dataset. It is calculated by averaging the precisions obtained for each category i within a total of C categories, expressed as follows:

$$mPA = \frac{\sum_{i=1}^{C} AP_i}{C} \quad (3)$$

In the context of this study, since four types of residues are considered, it is established that C=4. Precision, on the other hand, is defined as the proportion of samples that the model has correctly classified as belonging to a specific category in relation to the total number of samples identified. On the other hand, accuracy measures the proportion of correct predictions in the total number of cases; their respective formulas are presented below:

$$Precision = \frac{TP}{TP+FP} \quad (4)$$

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (5)$$

In the formulas, TP (True Positives) represents the number of correctly identified instances, while FP (False Positives) indicates instances incorrectly classified as positive when they are not. Similarly, FN (False Negatives) refers to instances that the model failed to correctly identify as positive, wrongly labelling them as negative. The recall, in turn, evaluates the proportion of samples that belong to a given category and have been properly recognized by the model. This metric is expressed as follows:

$$Recall = \frac{TP}{TP+FN} \quad (6)$$

Additionally, the F1 score is the harmonic mean of Precision and Recall and is calculated using the following formula:

$$F1\ score = 2 \cdot \frac{Precisión \cdot Recall}{Precisión + Recall} \quad (7)$$

To evaluate the trained model on the test set, the trainer. The test command is used, specifying the model best_model, the test data loader test_data, and a list of test metrics. In this case, DetectionMetrics_050 is used, setting score_thres to 0.5 and allowing a maximum of 50 predictions per image. Additionally, normalize_targets is enabled, and a post_prediction_callback is implemented to tune parameters for subsequent predictions, including a score_threshold of 0.5, an nms_top_k of 30, max_predictions of 30, and an nms_threshold of 0.7.

4. Results and Discussions

4.1. Progress of Evaluation Metrics

487 gradient updates are performed during each epoch, suggesting a constant fine-tuning process that avoids significant fluctuations and optimizes model performance without compromising convergence. Figure 6, developed in TensorBoard, shows the progress of mAP, precision, F1 score, and recall in each epoch.The mAP graph shows a continuous improvement trend throughout the 20 epochs. It is 0.65 in the second epoch but progressively increases until reaching 0.97 in the last epoch. This behavior indicates that, as the model progresses in training, it significantly improves its ability to detect recyclable solid waste. The upward trend shows that the model is learning effectively and adjusting its predictions throughout the process.

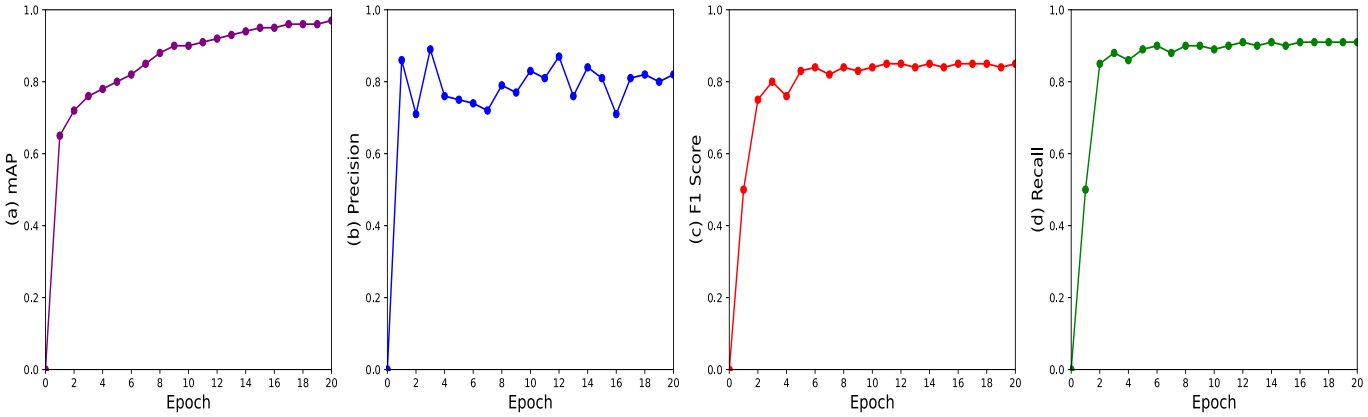

Fig. 6 Performance of the HuYOLO-NAS model during validation (a) mPA, (b) Accuracy, (c) F1 Score, (d) Recall

Regarding precision, the behavior is different. Precision reaches a maximum value of 0.90 in epoch 20. Throughout training, precision experiences some fluctuations but maintains an overall upward trend. Fluctuations in the middle epochs may indicate difficulties in avoiding false positives, but the continued improvement towards the end suggests that the model is managing to stabilise its ability to correctly identify recyclable waste. The F1-score follows a similar trend to the accuracy. In the beginning, the F1-score is low, but it gradually increases to around 0.85 in the last epoch. This reflects that, despite the initial fluctuations, the model improves its balance between accuracy and sensitivity as it progresses through training. The steady upward trend towards the end suggests that the model is effectively optimising both accuracy and sensitivity. The recall remains at high levels throughout training, starting at 0.00 in the first epoch and reaching 0.91 at epoch 20. Although it exhibits some small variations between epochs, the recall shows an overall positive trend. This indicates that the model has been consistent in its ability to correctly identify recyclable waste, reflecting its reliability in detecting relevant objects.

4.2. Analysis of Inference Time and Learning Curve of the HuYolo-NAS Model

Figure 7 presents the progress of the inference time and learning rate of the HuYOLO-NAS model over 20 iterations. First, the learning rate shows an evolution pattern that starts at extremely small values and increases sharply until reaching a value close to 4.8e-4 (0.00048) at iteration 4. Subsequently, the learning rate decreases steadily, reaching a value of approximately 5e-5 (0.00005) at iteration 19. This behavior is characteristic of the "one-cycle learning rate schedule", a strategy that seeks to balance initial exploration with subsequent fine-tuning to improve model efficiency and

prevent overfitting by avoiding premature convergence. As for the inference time during training (train_inference_time), the graph shows an evolution of time expressed in minutes. The inference time experiences an overall decrease from approximately 7 minutes and 50 seconds (470,000 ms) to 7 minutes and 25 seconds (445,000 ms) over the 20 iterations. Although there is a downward trend, fluctuations can be observed throughout the process, such as an increase in the inference time at iteration 7, followed by several minor ups and downs. Towards the end of the iterations, the time seems to stabilize, albeit with a small spike. These results suggest that although the model optimizes the process as training progresses, certain adjustments and changes in the environment or the model can influence the fluctuations in the inference time at some epochs.

4.3. Analysis of the Losses of the HuYolo-NAS Model

Figure 8 presents a detailed analysis of the training and validation losses of the HuYOLO-NAS model over the 20 epochs. In the overall loss, the reduction is notable: in the training set, the loss drops from 2.3 to 1.76, while in the validation set a decrease is also observed, from 2.4 to 1.65. However, fluctuations in validation are more pronounced, suggesting that the model has difficulties in generalising to new data, although it maintains a general trend of improvement. In classification, the training loss decreases from 1.2 to 0.88, reflecting an improved ability to classify residuals correctly. In validation, the classification loss decreases from 1.3 to 0.82 but with more oscillations, indicating that the model's ability to generalise across classes varies more when faced with unseen data.

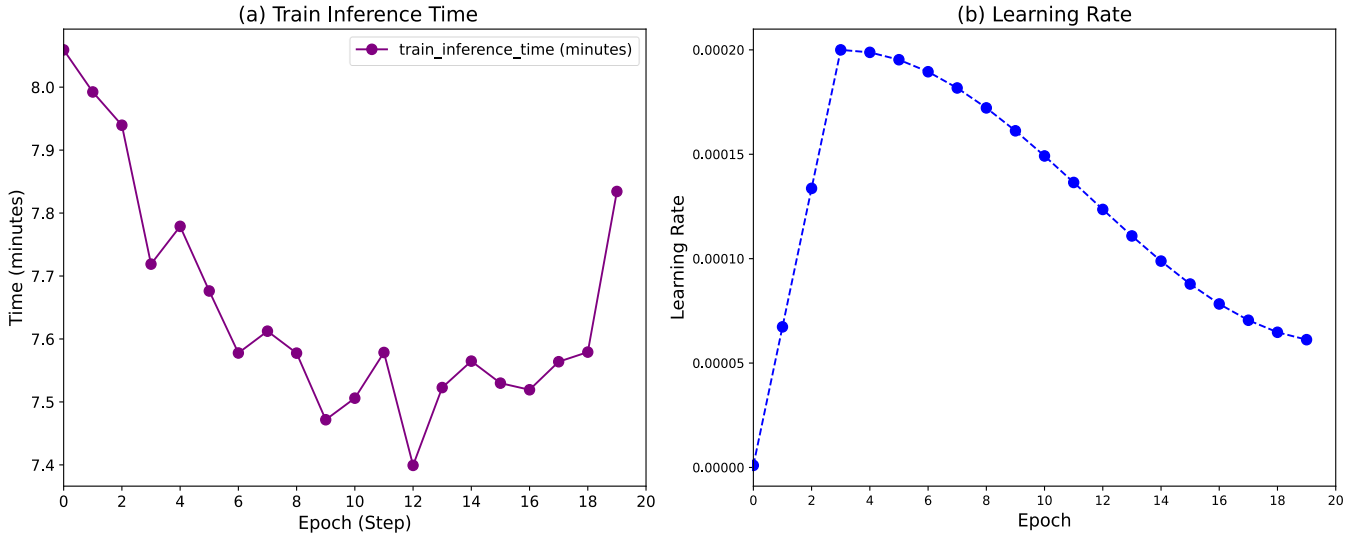

Fig. 7 Inference time and learning rate of HuYOLO-NAS: (a) Train Inference Time, (b) Learning Rate

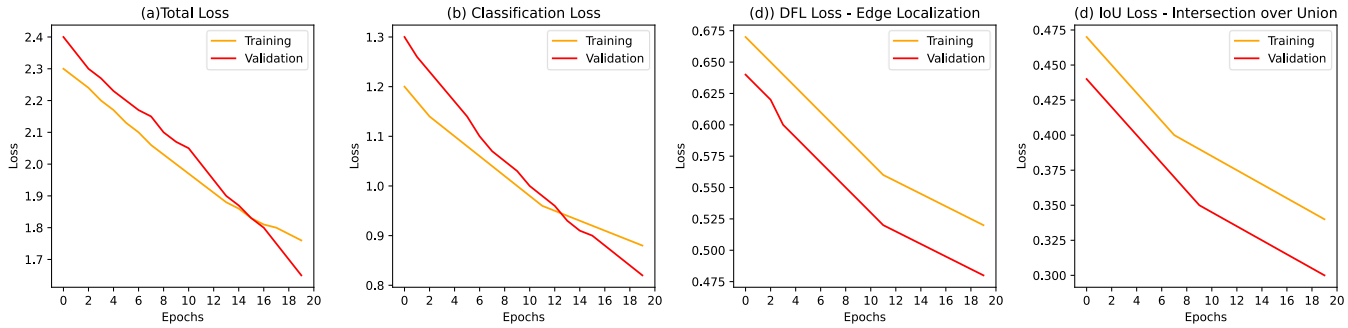

Fig. 8 Loss analysis in training and validation: (a) Total Loss, (b) Classification Loss, (c) DFL Loss, (d) IoU Loss

In terms of edge localisation (DFL), the loss in training shows a steady improvement from 0.67 to 0.52, and in validation, the loss decreases from 0.64 to 0.48, although also with fluctuations. These oscillations in validation suggest that the model is more unstable when trying to locate edges when faced with validation data, which could be improved with adjustments in regularisation or data augmentation. Finally, in the Intersection over Union (IoU) metric, the loss also decreases in both datasets: in training, it goes from 0.47 to 0.34, while in validation, it decreases from 0.44 to 0.30. Although some fluctuations are observed, the overall trend of improvement indicates that the model is achieving higher accuracy in object localisation, which is crucial for practical applications such as sorting recyclable waste.

Table 9. Values of the metrics in the validation of the HuYOLO-NAS model

Performance	Valor
Loss of Classification	0.7643
Loss IoU	0.2895
Loss DFL	0.4555
Total Loss	1.5094
Precision	0.7729
Recall	0.9445
mAP	0.9320
F1 Score	0.8374

Note: The values reflect the performance of the HuYOLO-NAS model during the validation process

4.4. Evaluation in the Validation of the HuYOLO-NAS Model

In the test performed (Table 9), the results of the HuYOLO-NAS model reflect outstanding performance in various detection metrics. The precision of 77.29% of the predictions made was correct, suggesting a good ability to identify relevant objects with a low percentage of false positives. This is particularly important in classification tasks, where high precision ensures that the detected objects are relevant. On the other hand, the model detected 94.45% of all objects present, highlighting its excellent ability to identify the most relevant elements without leaving many false negatives. This balance between precision and recall shows that the model has a high level of sensitivity, which is crucial for real-time applications that require high object detection without overlooking important elements. The mAP of 0.9320 demonstrates that the model maintains a good balance between precision and recall, which is critical for accurate object detection across a range of IoU thresholds. The mAP at 0.50 of 0.9320 demonstrates that the model maintains a good balance between accuracy and recall, which is critical for accurate object detection over a range of IoU thresholds.

This value reflects that the model not only correctly detects many objects but also has a high accuracy rate in its predictions. The F1 Score of 0.8374 reinforces this conclusion, showing that the model achieves an excellent combination of accuracy and sensitivity. Regarding localisation, the classification loss of 0.7643 and IoU loss of 0.2895 suggest that, although the model performs well in object identification and localisation, there is room for further optimisation in these areas, especially in the accuracy of bounding boxes.

The DFL loss of 0.4555 reflects the model's ability to correctly predict solid waste centre points, and the overall loss of 1.5094 shows satisfactory overall performance. Overall, these metrics highlight the model's effectiveness in detecting and locating recyclable waste. However, there is still room for improvement in specific areas, such as classification and accurate edge location. In addition to the key metrics, the Best Score Threshold of 0.7400 indicates the optimal threshold the model uses to classify detections, which is essential to adjust the balance between accuracy and recall, depending on the task's requirements. This value suggests that the model is tuned to maximise its performance, obtaining a good compromise between avoiding false positives. In that sense, the loss metrics, such as classification loss and IoU loss, together with the F1 score, show tha the HuYOLO-NAS model is well balanced in terms of efficiency and accuracy in the classification and localisation of solid recyclable waste.

4.5. Confusion Matrix of the HuYOLO-NAS Model

Analysis of the metrics obtained in the classification of recyclable solid waste with the HuYolo-NAS model reveals uneven performance between the different classes, highlighting both the strengths and areas for improvement of the model. In the Cardboard class, the accuracy is remarkably high, with a value of 0.9897, meaning that most of the predictions made as positive are correct. However, the recall is lower (0.8271), indicating that the model omits a significant number of cardboard residuals, with 60 false negatives. Although the model has a good hit rate, it does not detect all residuals in this class, a major limitation in its performance.

The F1 score of 0.9011, which measures the balance between accuracy and recall, is a favourable value and suggests that, despite the false negatives, the model is still performing well. However, improving recall in this class could increase the model's effectiveness without sacrificing accuracy. The model shows an outstanding performance for the Hard Plastic class, with a recall close to 100% (0.9935) and only 2 false negatives, showing that most hard plastic waste is correctly identified. The accuracy of 0.9747 is also high, although there are 8 false positives, suggesting that the model sometimes incorrectly classifies materials from other classes, such as hard plastics. Despite these false positives, the F1 score of 0.9840 is very high, confirming that the model performs exceptionally well classifying this class.

However, reducing the false positives could further improve the accuracy without losing the ability to detect the class effectively. For the Paper class, the model achieves an accuracy of 0.9934, implying that positive predictions for paper are generally correct. The recall of 0.9494 is also quite high, meaning the model correctly identifies most paper waste, although 16 false negatives persist. The F1 score of 0.9709 reflects a good balance between precision and recall, suggesting that the model adequately handles this class. However, as with the Cardboard class, improving the recall rate could help increase the model’s accuracy without compromising precision. In the Pet Plastic class, the model presents an accuracy of 0.9286, indicating a reliable classification but with room for improvement. On the other hand, the recall is lower (0.7948), meaning that the model is missing a significant amount of PET plastic waste, with 47 false negatives. Although the F1 score of 0.8565 is acceptable, this value reflects that the model could improve in terms of

detecting this class. The low recall is a concern, as it suggests that the model does not identify all PET waste despite its high accuracy. To improve performance in this class, it would be advisable to adjust the classification thresholds or implement additional techniques that allow for better isolation of PET plastic waste. Finally, the global mAP (Mean Average Precision) of 0.4472 reflects a moderate model performance overall. Although classes such as Hard Plastic and Paper perform outstandingly, the Pet Plastic class negatively affects the global average, underlining the disparity in performance between classes. The mAP is a measure that highlights the model's ability to correctly classify in a multi-class context. However, the moderate average suggests that despite good results in some classes, the model needs to be fine-tuned in the lower-performing classes to achieve greater homogeneity in its predictions.

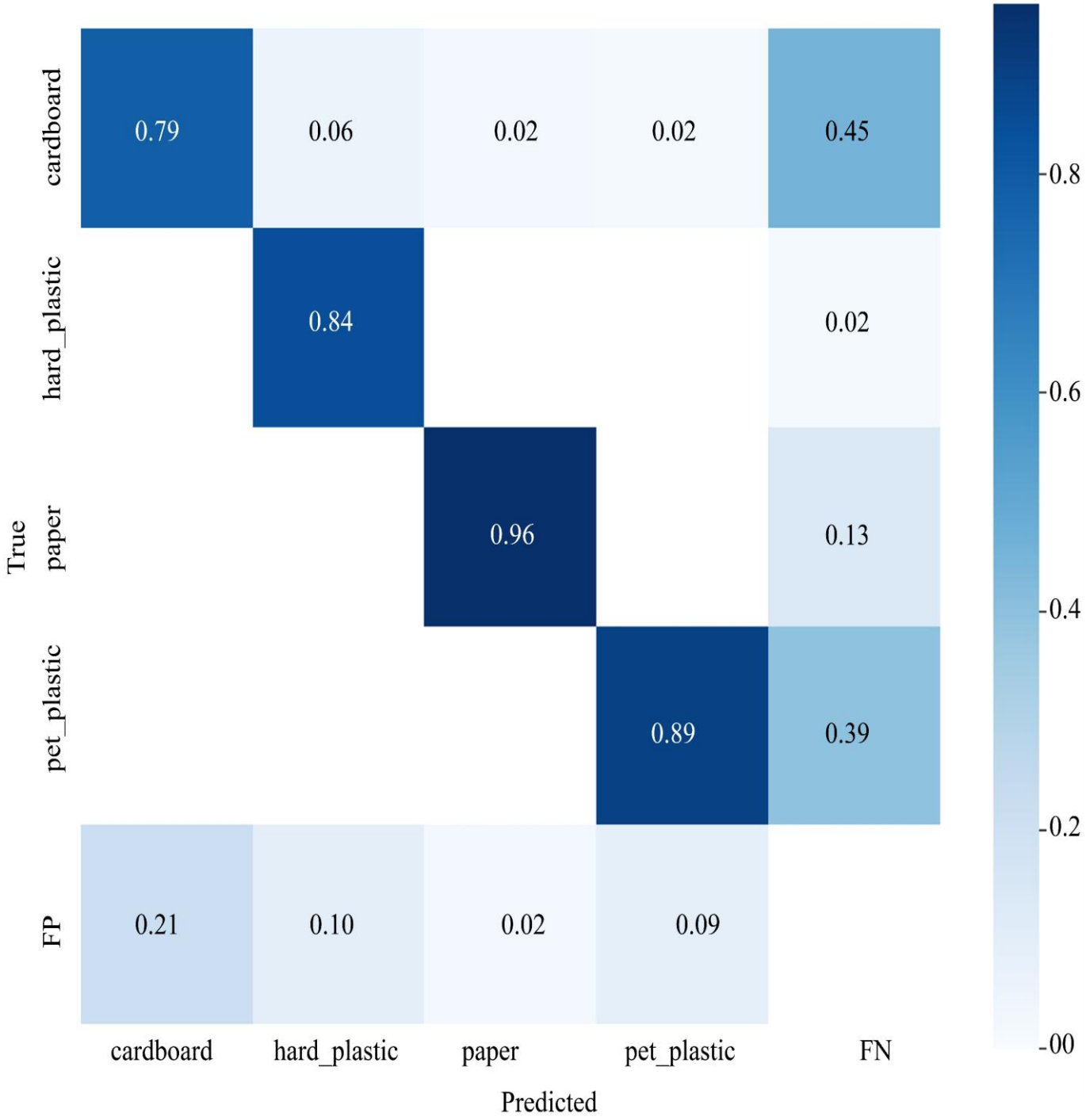

Fig. 9 Confusion matrix of the HuYolo-NAS model

The confusion matrix reveals that, while the model is highly accurate in its predictions, it has notable shortcomings in the sensitivity of certain classes. Specifically, 60 real instances are missed in cardboard, resulting in a recall of 82.71% despite near-perfect accuracy (98.97%), suggesting that the model is conservative in labelling this class and could benefit from less stringent thresholds. In Hard Plastic, detection is almost flawless, with only 8 false positives and a recall of 99.35%, indicating an outstanding ability to identify this category without confusing it with others. On the other hand, the Paper class shows a robust balance, although the 16 false negatives reflect a slight loss in complete detection, at a recall of 94.94%.

The greatest concern arises in the Pet Plastic class, where 47 instances are missed, resulting in a recall of 79.48% and evidencing that the model fails to capture a significant portion of this class despite a reasonable accuracy of 92.86%. These findings suggest that to improve overall performance, adjusting thresholds and optimising pre-processing, especially for Cardboard and Pet Plastic, is essential to achieve greater homogeneity in detecting recyclable waste.

Table 10. Performance of HuYolo-NAS in multi-label classification of recyclable solid waste

Class	Precisions (%)	Recall (%)	F1 Score (%)	Exactitude (%)	mPA (%)
Cardboard	98.97	82.71	90.11	82.55	90.11
Hard Plastic	97.47	99.35	98.40	97.22	98.40
Paper	99.34	94.94	97.09	95.01	97.09
Pet Plastic	92.86	79.48	85.65	83.13	85.65
Overall	92.41	88.12	92.06	89.23	92.06

Note: This table presents the results of precision, recall, F1 score, accuracy and mPA for each class of residues in the HuYolo-NAS model

The confusion matrix shows that the model incurs 60 false negatives in the Cardboard class, resulting in a recall of 82.71% despite an accuracy of 98.97%, indicating that, although the positive predictions are almost always correct, real instances of cardboard are missed. In Hard Plastic, 8 false positives are reported, allowing for a near-perfect recall of 99.35% and an accuracy of 97.47%, evidencing very robust detection with minimal confusion. The Paper class registers 16 false negatives, reflecting a solid balance with a recall of 94.94% and an accuracy of 99.34%, while Pet Plastic has 47 false negatives, resulting in a recall of 79.48% and an accuracy of 92.86%. These results indicate that the model has shortcomings in detecting instances in the Cardboard and Pet Plastic classes, suggesting the need to adjust the thresholds and improve the preprocessing to increase the sensitivity and homogeneity of the predictions. In conclusion, although the HuYolo-NAS model performs well overall, with outstanding performance in some classes, there are clear areas for improvement, particularly in the Pet Plastic class. Threshold adjustment, additional data processing or refinement in detecting specific classes could significantly improve the model's performance, raising both its accuracy and recall in the lower-performing classes. Next, we proceed to inference with the trained model, using the Supervision library to manage and process the model's detections.

4.6. Comparison of Metrics between Classes of Solid Waste

Table 10 shows clear differences in the performance of the HuYolo-NAS model between the residue classes, which allows the identification of specific areas for improvement. In the Cardboard class, the accuracy is very high (98.97%), meaning that almost all instances identified as cardboard are correct; however, the recall (82.71%) indicates that the model misses approximately 17% of the real cases, suggesting that thresholds should be adjusted or more examples incorporated to reduce false negatives in this category. On the other hand, the Hard Plastic class achieves an accuracy of 97.47% and an almost perfect recall (99.35%), which is reflected in an F1 score of 98.40%, evidencing almost error-free detection in terms of both false positives and false negatives.

The Paper class shows a solid balance, with an accuracy of 99.34% and a recall of 94.94%, yielding an F1 score of 97.09%; these results indicate that the model manages to capture most paper instances with very few errors. In contrast, the Pet Plastic class has an accuracy of 92.86%, but its recall is low (79.48%), resulting in an F1 score of 85.65%. This means that, although Pet Plastic's predictions are largely correct, the model fails to detect a considerable proportion of instances, negatively impacting this class's overall performance. Overall, the global averages (precision 92.41%, recall 88.12%, F1 score 92.06% and accuracy 89.23%) reflect robust system performance. However, the variability between classes indicates the need to optimise the model, especially for Pet Plastic detection and, to a lesser extent, for cardboard, where increasing recall could significantly improve overall effectiveness without compromising the high accuracy achieved in the other categories.

To visualise the set of predicted images, sv.DetectionDataset.from_yolo loads the dataset in YOLO format, accessing the images and their annotations; subsequently, the best_model.predict variable makes predictions on each image by applying a confidence threshold to limit the relevant detections, while sv.Detections organise the coordinates of the bounding boxes, the confidence and the detected classes. Finally, the Supervision libraries are used to manage the annotations and predictions, NumPy to adjust the coordinates, OpenCV to draw and label the predictions on the images, and Matplotlib to visualise them (Figure 10).

Fig. 10 Recyclable solid waste detection results of the HuYolo-NAS model

4.7. Comparison of Performance in Different Dimensions of the Dataset

Table 11 shows the performance of the HuYOLO-NAS model in the multi-label classification of solid recyclable waste using different input resolutions, with a partition of 70% for training, 10% for validation and 10% for testing. The average values of five key metrics are presented: precision, recall, F1 Score, accuracy and mPA. Each metric is evaluated to verify model performance at different image sizes, which allows for identifying the optimal resolution to maintain a balance between detection quality and computational efficiency.

The 416x416 and 320x320px resolution show superior accuracy, reaching an average of 53.41%. This suggests that both resolutions have a lower number of false positives compared to the larger resolutions. This behaviour is appropriate where incorrect detections need to be minimised, as higher accuracy indicates a higher proportion of correct predictions among all positive detections.

Table 11. Overall performance of HuYOLO-NAS at different resolutions for multi-label classification of recyclable solid waste

Resolution (px)	Precision (%)	Recall (%)	F1 Score (%)	Accuracy (%)	mPA (%)
640x640	52.78	72.54	60.79	79.92	63.34
512x512	49.12	72.15	58.57	79.43	62.88
416x416	53.41	73.52	61.64	80.45	63.67
320x320	53.41	73.52	61.64	80.45	63.67

Note: This table presents the average results of precision, recall, F1 score, accuracy, and mPA for the HuYOLO-NAS model at different input resolutions of 640x640px, 512x512px, 416x416px, and 320x320px.

The recall metric is slightly higher for the 416x416 and 320x320 datasets, reaching a value of 73.52%. This indicates that these resolution sizes allow capturing a higher number of true positives compared to the 640x640 and 512x512 resolutions, resulting in improved object detection coverage. In classification tasks, a high recall is desirable, as it ensures that the model identifies the most positives, reducing the risk of false negatives. The 416x416 and 320x320 resolutions also achieve the highest F1 Score with 61.64%.

This indicator reflects a good balance between accuracy and recall, which is fundamental in classification tasks where both aspects are critical. A high F1 Score suggests that these resolutions allow for a balanced performance between minimising false positives and false negatives, which is key for effective and reliable classification. In terms of accuracy, the 416x416 and 320x320px resolutions again stand out, with an average of 80.45%. This value shows that these resolutions have a higher hit ratio compared to larger resolutions, implying a higher level of correct predictions for both true positives and true negatives. Therefore, these resolutions could be a preferred option when seeking to maximise the overall correct prediction rate.

The mPA metric also follows a similar pattern, with the highest values obtained for the 416x416 and 320x320 resolutions, reaching 63.67%. This metric measures the average accuracy per pixel and, in this case, highlights the effectiveness of these resolutions in tasks where detailed evaluation at the pixel level is important. The high mPA indicates that the model maintains solid performance in granular-level accuracy, which is relevant in segmentation or exhaustive detection tasks. Overall, 416x416 and 320x320px datasets offer slightly better and more balanced performance in almost all metrics evaluated, compared to 640x640 and 512x512 resolutions. This analysis suggests that, for this specific task, a medium resolution (such as 416x416 or 320x320) could be more efficient and effective, providing an optimal balance between precision, recall, F1 Score and accuracy. Furthermore, opting for these resolutions could reduce the computational cost, making the model more viable in terms of processing and storage without compromising classification performance.

4.8. Implementation of the Code with the Robotic Arm

Once the accuracy of the HuYolo-NAS model was corroborated, it was implemented in conjunction with a robotic arm in charge of moving the identified solid waste to the corresponding containers. Figures 12, 13, 14, 15 and 16 show an example of the functional robot in action, carrying out the transfer of a box.

4.9. Advantages of HuYOLO-NAS Over Advanced Machine Vision Techniques

The proposed HuYOLO-NAS demonstrated superior performance compared to state-of-the-art techniques reported in the state of the art, such as DSYOLO-trash [17], GCDN-Net [18] and methods based on ShuffleNet v2 enhanced with Yolov5s [19]. This model integrated the YOLO-NAS architecture with the Hu Moments algorithm, resulting in three-dimensional coordinates invariant to geometric transformations and, thus, more accurate spatial localisation of recyclable waste in highly variable environments.

The rigorous pre-processing and labelling of the EcoSight dataset, consisting of 8,400 images from automated and private sources, facilitated robust pattern learning, which was reflected in a mAP of 93.20% and a recall of 94.45%. These metrics show that the model minimises both false positives and negatives, overcoming the limitations of approaches that rely exclusively on deep feature extraction in controlled scenarios.

Furthermore, integrating the HuYOLO-NAS model with the HuArm robotic arm provided added value by allowing the translation of the detection into physical actions through accurate inverse kinematics calculations and real-time communication with the ESP32-WROOM-32E. This synergy between machine vision and robotics optimises the waste sorting and handling process, reducing human intervention and improving operational efficiency, unlike other techniques described in the literature that focus only on sensing without addressing the automated handling process. Overall, the combination of a robust adaptive architecture, a comprehensive pre-processing process and integration with a functional robotic system allows HuYOLO-NAS to outperform existing techniques, offering a scalable and efficient solution for sorting solid recyclable waste in high variability environments.

4.10. Implementation Challenges

Implementing the HuYolo-Nas model with the HuArm presented several challenges during its development. The model training time, coupled with the need for a varied image database, presented a greater challenge than anticipated. It was necessary to expand the dataset by capturing new images to improve the effectiveness of the HuYolo-Nas model, which prolonged the process. In addition, equipment with adequate performance and a stable training environment were required to avoid interruptions and minimise detection errors.

The design of the HuArm underwent multiple modifications due to the diversification of the servomotors used to control each degree of freedom, as well as the need to ensure sufficient reach distance to move the solid recyclable waste from the starting point to the final deposit. This problem was mainly due to the absence of a specific reference for the optimal design of a claw-type robotic arm. Also, the interaction between the model and the arm was disrupted due to the lack of a freely programmable and low-cost control device to establish a wireless connection with the computer to accurately transmit the movement coordinates.

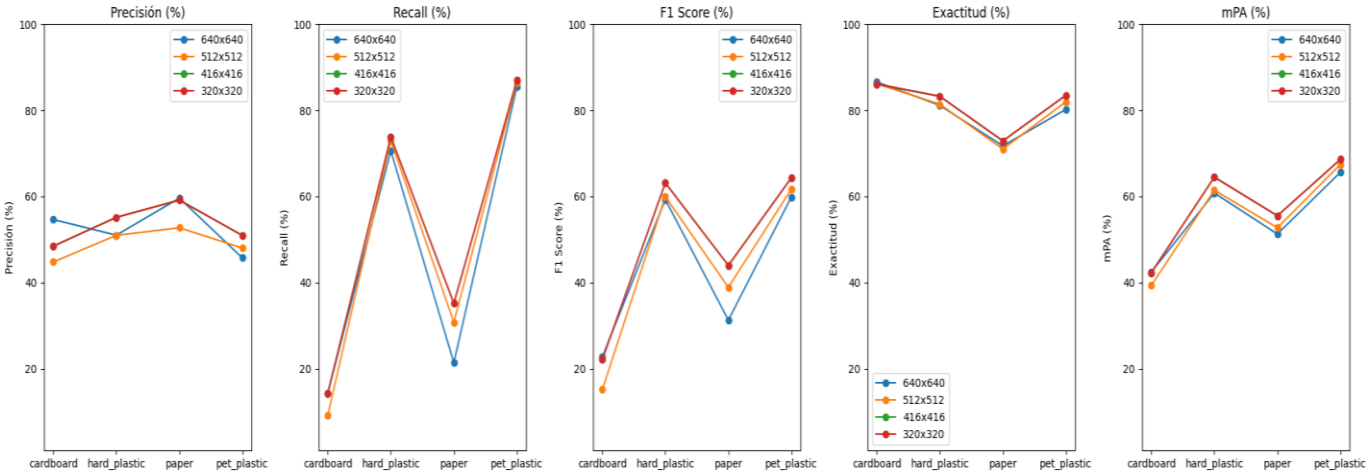

Fig. 11 Inference results of the HuYolo-NAS model

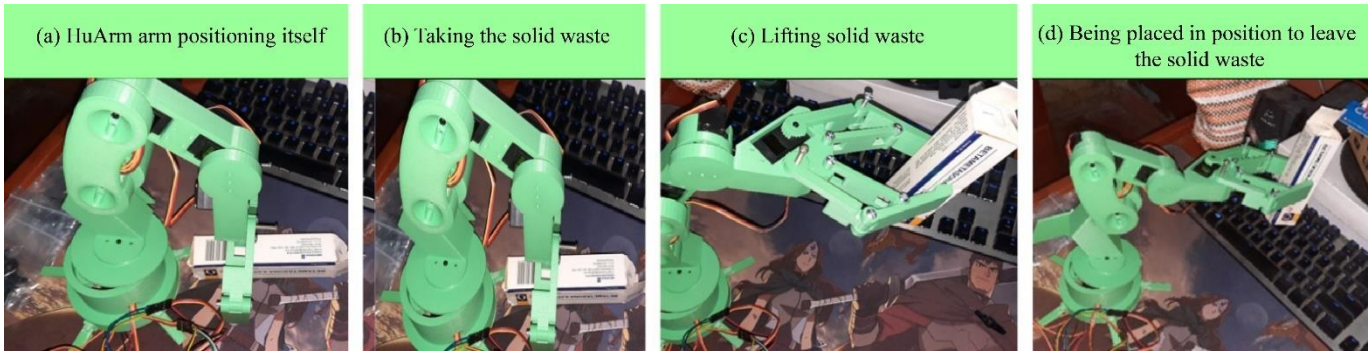

Fig. 12 HuArm operation: (a) HuArm arm positioning itself, (b) Taking the solid waste, (c) Lifting solid waste, (d) Being placed in a position to leave the solid waste

5. Conclusion

Throughout the training of the HuYOLO-NAS model, significant progress has been observed in the evaluation metrics, aligned with the purpose of classifying recyclable solid waste to optimize separation in recycling in highly variable environments. First, a progressive improvement was obtained in the mAP (Figure 6), which went from 0.7608 to 0.9504, indicating an increasingly precise recyclable solid waste detection capacity. This result suggests that the model has achieved an accurate classification of the four types of recyclable waste predicted in the dataset, fulfilling the objective of improving classification.

Regarding precision and F1-score, although in the first epochs, there were fluctuations (Table 4), at the end of training, consistent values were reached, with precision increasing from 0.7608 to 0.9504 and F1-score improving from 0.04011 to 0.07716. This performance reflects the model's ability to effectively learn and generalize recyclable waste, contributing to the optimization of the sorting process in a dynamic recycling environment. The recall (Figure 7) showed outstanding performance, increasing from 0.9748 to 0.9985, demonstrating that the model effectively detects almost all recyclable waste, even under high variability conditions. This is a key result in meeting the objective of accurate sorting. In the loss analysis (Figure 9), a steady decrease was observed in both sorting losses, DFL and IoU, reflecting an improvement in the model's ability to identify and locate solid waste. The total loss went from 2.35 to 1.829, while in validation, it decreased from 2.25 to 1.6 (Table 3), indicating an improvement in the model's generalized learning.

In terms of the tests performed with the test set, the model achieved an accuracy of 0.531 in the first test and 0.368 in the second, showing how the reduction in the score threshold and the variability in the NMS threshold affect accuracy (Figure 8). However, this adjustment favoured higher recall, which is valuable to ensure that the system classifies the majority of recyclable waste correctly. The mAP remained stable, with values ranging from 0.9522 to 0.9509 (Figure 5), reflecting that threshold reduction did not significantly compromise sorting ability.

The confusion matrix (Figure 8) revealed that some classes, such as hard_plastic, presented a low precision due to a high number of false positives, but the pet_plastic class showed the best performance, with a precision of 0.8753 and

accuracy of 0.8833 (Table 6). This indicates that the model is more effective in classifying certain classes, which will allow future optimizations to improve performance in less accurate classes. In the performance analysis at different resolutions, it was observed that resolutions such as 416x416 and 320x320 (Figure 11) provide a good balance between accuracy, recall, and computational performance. These sizes achieved a mAP of 63.67% (Table 7), suggesting that an intermediate resolution optimises both detection capability and computational efficiency, which is crucial in dynamic recycling environments where time and resources are limited.

Furthermore, the findings of this study are closely linked to existing waste management policies, particularly the Integrated Solid Waste Management Law enacted in Peru [1]. Implementing technologies such as HuYOLO-NAS supports government efforts to improve waste separation and recycling efficiency, which is crucial to promoting sustainable resource management. The system's ability to operate in highly variable environments reinforces the need to adopt innovative solutions that leverage current policies, thus facilitating the transition towards more sustainable waste management models. Finally, the HuYOLO-NAS model has proven to be effective in sorting recyclable solid waste, achieving high levels of accuracy and recall, and showing optimal performance in environments with high variability. The analysis also highlights the importance of tuning the model parameters and resolution to maximise its efficiency, which meets the research objectives.

5.1. Recommendations

To improve the model's performance in classifying recyclable waste, it is recommended to adjust the classification threshold, especially for classes with low recall, such as Pet Plastic. This will help reduce false negatives and improve detection. In addition, it would be useful to implement data augmentation techniques to increase the variability of these classes and strengthen their recognition.

Regarding object localization, it is necessary to continue optimizing the hyperparameters to reduce fluctuations in the localization metrics and improve accuracy in the bounding boxes. This is key to avoiding errors in detecting small or partially covered objects. It is also observed that, although inference times have improved, fluctuations continue to exist. Additional optimisation techniques, such as quantization or pruning, are recommended to reduce these variations and improve consistency in processing time. Finally, it would be beneficial to continue enriching the dataset, particularly with under-represented classes, such as Cardboard and Paper, to improve the generalisability of the model in various scenarios. Diversification and expansion of the data will allow the model to better adapt to variations in the conditions of the recycling environment.

References

[1] Grese Hanampa et al., "Occupational Safety in the Management of Pet Plastic Waste Generated by the Lot 57 Camp in Camisea and Analysis of Economic Losses Due to Poor Segregation," *Chemical Engineering Transactions*, vol. 111, pp. 541-546, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[2] Jorge Cristóbal et al., "Climate Change Mitigation Potential of Transitioning from Open Dumpsters in Peru: Evaluation of Mitigation Strategies in Critical Dumpsites," *Science of the Total Environment*, vol. 846, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[3] Annie Lopez-Yamunaqué, and Jose Alberto Iannacone, "Integral Management of Urban Solid Waste in Latin America," *Paideia XXI*, vol. 11, no. 2, pp. 453-474, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[4] Angela María Pacheco, Iván Darío Porras, and Daniel Alejandro Rodríguez, "Device for Classifying Solid Waste and Measuring Ecological Footprint," *Habitus Magazine: Research Hotbeds*, vol. 1, no. 2, pp. 1-17, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[5] Armando Vélez-Azañero, Naty Luque Sandoval, and Daniela Vilchez Aguilar, "Solid Waste from the Lurín River Basin, Lima, Peru," *Amazonian Science*, vol. 9, no. 2, pp. 81-92, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[6] Yoni Mateo Valiente Saldaña et al., "Municipal Solid Waste Management in the District of Trujillo, Peru," *Venezuelan Management Magazine: RVG*, vol. 28, no. Extra 10, pp. 1527-1540, 2023. [Google Scholar] [Publisher Link]

[7] Nelly Sara Chacaltana Lara, "Solid Waste Recovery in Peru," *Journal of Climatology Special Edition Social Sciences*, vol. 23, pp. 3281-3290, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[8] E.J. Preciado Jeronimo, and E.A. Lara Medina, "*Main Deficiencies in the Implementation of the Solid Waste Management Policy in Lima and Callao*," University of the Pacific, Lima, Peru, 2022. [Google Scholar]

[9] Harald Ian D.I. Muri, and Dag Roar Hjelme, "Classification of Municipal Solid Waste Using Deep Convolutional Neural Network Model Applied to Multispectral Images," *Automated Visual Inspection and Machine Vision IV*, vol. 11787, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[10] Lynda Andeobu, Santoso Wibowo, and Srimannarayana Grandhi, "Artificial Intelligence Applications for Sustainable Solid Waste Management Practices in Australia: A Systematic Review," *Science of The Total Environment*, vol. 834, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[11] A.V. Shreyas Madhav et al., "Application of Artificial Intelligence to Enhance Collection of E-Waste: A Potential Solution for Household WEEE Collection and Segregation in India," *Waste Management & Research: The Journal for a Sustainable Circular Economy*, vol. 40, no. 7, pp. 1047-1053, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[12] Judith Soledad Yangali Vicente et al., "Ecological Behavior and Environmental Culture, Promoted Through Virtual Education in Students From Lima-Peru," *Journal of Social Sciences*, vol. 27, no. 1, pp. 1-15, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[13] Ana Teresa Mosquera Espinosa**,** and Laura Juliana Caro Moreno, *Alternatives for Solid Waste Management and its Integration in the Assembly of an Agroecological Garden*, 1st ed., Javeriano Publishing House, 2019. [CrossRef] [Publisher Link]

[14] James G. Shanahan, and Liang Dai, "Introduction to Computer Vision and Real Time Deep Learning-based Object Detection," *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, Virtual Event, CA, USA, pp. 3523-3524, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[15] Charu C. Aggarwal, *Neural Networks and Deep Learning*, 2nd ed., Springer Cham, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[16] Argimiro Arratia, Alejandra Cabaña, and José Rafael León, "Deep and Wide Neural Networks Covariance Estimation," *Artificial Neural Networks and Machine Learning – ICANN 2020: 29th International Conference on Artificial Neural Networks*, Bratislava, Slovakia, pp. 195-206, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[17] Howard B. Demuth et al., *Neural Network Design*, 2nd ed., Martin Hagan, Stillwater, OK, United States, pp. 1-800, 1996. [Google Scholar] [Publisher Link]

[18] Wanqi Ma et al., "DSYOLO-Trash: An Attention Mechanism-Integrated and Object Tracking Algorithm for Solid Waste Detection," *Waste Management*, vol. 178, pp. 46-56, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[19] Md Mosarrof Hossen et al., "GCDN-Net: Garbage Classifier Deep Neural Network for Recyclable Urban Waste Management," *Waste Management*, vol. 174, pp. 439-450, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[20] Yujin Chen et al., "Classification and Recycling of Recyclable Garbage Based on Deep Learning," *Journal of Cleaner Production*, vol. 414, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[21] Xiaohang Yang et al., "Adaptive Neural Network Control of Manipulators with Uncertain Kinematics and Dynamics," *Engineering Applications of Artificial Intelligence*, vol. 133, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[22] Sergio Ferrarini et al., "A Method for the Assessment and Compensation of Positioning Errors in Industrial Robots," *Robotics and Computer-Integrated Manufacturing*, vol. 85, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[23] Dimitrios Stamoulis et al., "Designing Adaptive Neural Networks for Energy-Constrained Image Classification," *International Conference on Computer-Aided Design*, San Diego, CA, USA, pp. 1-8, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[24] Eduardo De-La-Torre-Jave, Aldo Alvarez-Risco, and Shyla Del-Aguila-Arcentales, *Circular Economy and Recycling in Peru*, Towards a Circular Economy, Springer, Cham, pp. 281-295, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[25] Hui Zhang et al., "A Method to Create Training Dataset for Dehazing with Cyclegan," *IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium*, Waikoloa, HI, USA, vol. 521, pp. 6997-7000, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[26] Adwait Arayakandy, Ankit Gupta, and Rashmi Thakur, "Design and Development of Classification Model for Recyclability Status of Trash Using SVM," *International Journal for Research in Applied Science and Engineering Technology*, vol. 7, no. 3, pp. 2146-2150, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[27] László Moldvai, "Weed Detection and Classification with Computer Vision Using a Limited Image Dataset," *Applied Sciences*, vol. 14, no. 11, pp. 1-26, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[28] Kerr Fitzgerald et al., "Multi-Resolution Fine-Tuning of Vision Transformers," *Medical Image Understanding and Analysis: 26th Annual Conference*, Cambridge, UK, pp. 535-546, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[29] Lin Chengchuang et al., "Review of Image Data Augmentation in Computer Vision," *Journal of Computer Science and Technology*, vol. 15, no. 4, pp. 583-611, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[30] Bart Degraeuwe, "Prioritising the Sources of Pollution in European Cities: Do Air Quality Modelling Applications Provide Consistent Responses?," *Geoscientific Model Development*, vol. 13, no. 11, pp. 5725-5736, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[31] Ming-Kuei Hu, "Visual Pattern Recognition by Moment Invariants," *IRE Transactions on Information Theory*, vol. 8, no. 2, pp. 179-187, 1962. [CrossRef] [Google Scholar] [Publisher Link]

[32] Juan Terven, Diana-Margarita Córdova-Esparza, and Julio-Alejandro Romero-González, "A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS," *Machine Learning and Knowledge Extraction*, vol. 5, no. 4, pp. 1680-1716, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[33] Hicham Slimani, Jamal El Mhamdi, and Abdelilah Jilbab, "Deep Learning Structure for Real-Time Crop Monitoring Based on Neural Architecture Search and UAV," *Brazilian Archives of Biology and Technology*, vol. 67, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[34] Yao Lu et al., "Scaling Up Quantization-Aware Neural Architecture Search for Efficient Deep Learning on the Edge," *Proceedings of the 2023 Workshop on Compilers, Deployment, and Tooling for Edge AI*, Hamburg Germany, vol. 34, pp. 1-5, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[35] Chang Liu, and Dong Zhang, "A Selective Quantization Approach for Optimizing Quantized Inference Engine," *2023 11th International Conference on Information Systems and Computing Technology*, Qingdao, China, vol. 29, pp. 92-99, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[36] Xiangxiang Chu, Liang Li, and Bo Zhang, "Make RepVGG Greater Again: A Quantization-Aware Approach," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 10, pp. 11624-11632, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[37] Thanh-Truong Nguyen, and Cong Vo Duy, "Grasping Moving Objects with Incomplete Information in a Low-Cost Robot Production Line Using Contour Matching Based on the Hu Moments," *Results in Engineering*, vol. 23, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[38] Randhi Uma Devi et al., "A Sustainable Approach for An Integrated Municipal Solid Waste Management," *Recent Trends in Solid Waste Management*, pp. 55-74, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[39] Kevin M. Lynch, and Frank C. Park, *11- Robot Control*, Modern Robotics: Mechanics, Planning, and Control, Cambridge University Press, pp. 349-399, 2017. [CrossRef] [Google Scholar] [Publisher Link]