iscte

INSTITUTO
UNIVERSITÁRIO
DE LISBOA

Identification of residues deposited outside of the deposition equipment, using video analytics

Soraia Hermínia Aguiar Afonso Fernandes

Master Degree in Telecommunications and Computer Engineering

Supervisor:
PhD Tomás Gomes da Silva Serpa Brandão, Assistant Professor,
ISCTE-IUL

Co-Supervisor:
PhD Luís Miguel Martins Nunes, Associate Professor,
ISCTE-IUL

October, 2021

iscte

TECNOLOG|AS
E ARQU|TETURA

Department of Information Science and Technology

Identification of residues deposited outside of the deposition equipment, using video analytics

Soraia Hermínia Aguiar Afonso Fernandes

Master Degree in Telecommunications and Computer Engineering

Supervisor:
PhD Tomás Gomes da Silva Serpa Brandão, Assistant Professor,
ISCTE-IUL

Co-Supervisor:
PhD Luís Miguel Martins Nunes, Associate Professor,
ISCTE-IUL

October, 2021

Acknowledgements

I would like to thank the following people, without whom I would not have been able to have made it through this dissertation.

To my three supervisors Professor Tomás Brandão, Professor Luís Nunes and Professor João Carlos Ferreira for teaching me so much about machine learning and for their willingness for providing guidance and feedback that substantially improved the quality of this dissertation.

To my family, especially my parents for their constant support that they have shown me though this project.

This work was partly funded through national funds by FCT - Fundação para a Ciência e Tecnologia, I.P. under projects UIDB/EEA/50008/2020 (Instituto de Telecomunicações) and UIDB/04466/2020 (ISTAR)

To all those I have listed my sincere "Thank You".

Resumo

Nas áreas onde a produção de resíduos é excessiva, por vezes ocorre a deposição indevida em torno dos equipamentos de deposição de lixo, exigindo mais esforço por parte das equipas de recolha destes resíduos. Nesta dissertação é proposto um sistema de reconhecimento de imagem para a deteção e classificação de resíduos fora dos equipamentos de deposição existentes para o mesmo. A principal motivação é facilitar o trabalho de recolha dos resíduos na cidade de Lisboa. De forma a possibilitar o desenvolvimento de algoritmos que possam vir a ser úteis na automatização de tarefas em diferentes áreas de intervenção, a Câmara Municipal de Lisboa criou um repositório, denominado 'LxDataLab', contendo vários conjuntos de dados. Estes dados, por sua vez são submetidos a um processo pré-processamento e por fim são submetidas para deteção e classificação dos resíduos. Assim é proposto um método de classificação e identificação de resíduos utilizando redes neuronais para análise de imagens: a primeira abordagem consistiu no treino de uma rede neuronal convolucional de aprendizagem profunda (CNN) desenvolvida especificamente para classificar resíduos; numa segunda abordagem foi treinada uma CNN utilizando um modelo pré-treinado MobileNetV2. Nesta última abordagem, o treino foi mais rápido em relação à abordagem anterior, e o desempenho na deteção da classe e da quantidade de resíduos nas imagens foi superior. A taxa de acerto para as classes de resíduos selecionadas variou nos 80% para o conjunto de teste. Após a deteção e classificação dos resíduos nas imagens são geradas anotações nas mesmas.

Palavras-Chave: Redes Neuronais Convolucionais; Aprendizagem automática; Processamento de Imagem; Arquitetura neuronal.

Abstract

In areas where waste production is excessive, sometimes improper deposition occurs around the garbage equipment, requiring more effort from the waste collection teams. In this dissertation an image recognition system is proposed for the detection and classification of waste outside the existing waste disposal equipment. The main motivation is to facilitate the work of waste collection in the city of Lisbon, which is done by the teams of the Lisbon Waste Collection Centers. In order to help the waste collection planning, the collection team inspectors in partnership with the Lisbon City Council created a repository with several datasets, which they named, 'LxDataLab'. The collected images go through the pre-processing process and finally are submitted to waste detection and classification, through deep learning networks. In this sense, a classification and identification method using neural networks for image analysis is proposed: the first approach consisted in training a deep learning convolutional neural network (CNN) specifically developed to classify residues; in a second approach a CNN was trained using a pre-trained MobileNetV2 model, which only the last layer was trained. The training in this approach was faster compared to the previous approach, as were the performance values in detecting the class and the amount of residues in the images. The hit rate for the classification of the selected debris varied between 80%, for test set. After the detection and classification of the residues in the images are recognized, annotations are generated on the images.

Keywords: Convolutional neural networks; Machine learning; Image processing; Neural Architecture.

Index

Table of Contents

Table of Figures

Glossary of Abbreviations and Acronyms

AI	Artificial Intelligence
ANN	Artificial Neural Networks
CNN	Convolutional Neural network
DCNN	Deep Convolutional Neural Network
DL	Deep Learning
KNN	K-Nearest Neighbour
mAP	Mean Average Precision
ML	Machine Learning
PoC	Proof of Concept
SVM	Support Vector Machine
YOLO	You Only Look Once

Chapter 1 – Introduction

Urban solid waste, commonly known as garbage, can be defined as everything that is considered leftover from a specific product[1]. This category includes recyclable materials, organic waste, garden waste, and bulky waste. Its management has been one of the main challenges for Portugal, more specifically for municipalities and government officials.

In recent years, the increase of the worldwide population, together with a society that became consumerist, resulted in more production, more consumption and therefore a larger amount of produced waste, which translates into insufficient infrastructures for the collection and treatment of waste, thus causing great harm to the environment.

According to the 2018 Annual Report on Urban Waste[2], each Portuguese citizen generates an average of about 505kg of waste per year (well above the European average – 476kg/year). The report[1] also states that 5.2 million tons of urban waste were collected in Portugal (+21.1kg inhabitant/year of what was generated in 2017) which represents an increase of 4% over the previous year.

Efforts that aim to decrease the statistical values mentioned rely on increasing the percentage of recycling, the economic sustainability of the models that generate waste, and the decrease in the amount of waste that is disposed of in landfills.

Much of the generated waste, more specifically solid urban waste, is recyclable, which means that all the waste collected goes through the process that transforms used materials into new products. Depending on the type of waste, different recycling processes are followed, and therefore, applying methods that allow the correct disposal of waste in the equipment designated for this can bring benefits. The existing techniques that allow the separation of waste, more specifically the selective sorting (garbage recycling containers, glass), the models developed and the set of awareness campaigns in order to facilitate the

[1] From website: http://www.simar-louresodivelas.pt/Resi_urb_pag/recolha_urbanos.aspx

[2] From website: https://apambiente.pt/sites/default/files/_Residuos/Producao_Gest%C3%A3o_Residuos/Dados%20RU/RARU%202018.pdf

work of collection, have become essential, but still insufficient to reduce environmental impact.

Excessive garbage generation or insufficient frequency of garbage collection causes citizens to dispose of garbage outside the containers, so automatic detection of such situations can help the collection process.

In order to address this problem, Lisbon City Hall has ongoing strategies such as:

- Installation of underground recycling equipment, hoping to lessen the aesthetic impact that garbage generates in the streets. This type of equipment consists of larger waste containers, when they become full, people often deposit garbage in the vicinity of this equipment.
- Optimization of the waste collection circuits, with methods such as the installation of 1500 sensors[3] in several containers scattered around the city – this measure aims to identify how full the containers are. However, the use of these sensors does not provide information about the accumulation of garbage around the equipment.

City locations where the production of waste is excessive often lead to garbage disposal around the equipment, implying an increased effort to the collection teams assigned to those areas. In this sense, it is important to foresee actions and anticipate the scenarios.

Deep learning mechanisms have been used to implement systems capable of detecting waste through image recognition. The creation of a trained model to detect and classify waste placed outside the disposal equipment can improve the management of collection operation.

[3] From website: https://lisboainteligente.cm-lisboa.pt/lxi-iniciativas/sensorizacao-dos-depositos-coletivos-de-residuos/

1.1. Motivation

The waste collection operation management process is a complex and extensive one. Ensuring it is being done with quality is one of the main targets. However, this process will only succeed if people are motivated to perform the correct recycling of waste. Places where the production of waste is excessive often lead to people placing waste outside of the disposal equipment, because the equipment is already full or by sloppiness, laziness and lack of civility. In other cases, these residues are wrongly placed because people do not know where the correct place is, where they should be placed, in order to be recycled, as is the case of large-sized residues such as furniture and house appliances.

One of the main objectives of computer vision-based systems is to perform tasks that mimic the human visual system, namely the classification and detection of objects, and understanding the context in which the objects are found. However, there is a huge separation between what humans and computers 'see'. For computers to be able to see what humans see, they need an input, which is the form of images. In general, image processing often uses convolution methods to extract the main features. Which means performing multiple matrix multiplications with the matrix that represents the image. With these features it is possible at a later stage to perform detection and classification of objects in an image. In this sense, the waste collection operation could benefit from a Computer Vision based system that analyzes images depicting the vicinity of the waste disposal equipment to determine if there is waste placed outside the containers.

Such system could help the management team to monitor the amount of disposed waste in problematic locations, such as those located in Lisbon.

Currently collection management is performed door to door, and the resource of underground recycling bins has become the first approach to cover the visual pollution. However insufficient to supply the amount of waste produced by local people.

In recent years, deep learning applications based on convolutional neural networks have been applied quite successfully to image classification and object detection problems. However, training a deep learning-based model with sufficient accuracy for a given task implies the use of a robust dataset. In the case of misplaced garbage detection, there is a large variation for both the disposal equipment setup and waste types. This variation will therefore require a larger diversity of images for training the classification systems, in order to achieve an accuracy that will allow to identify the locations requiring an immediate action by the collection team.

1.2. Research Questions

Although the recognition of objects by the human brain is usually more accurate, today many computer systems already play the same role as humans, guaranteeing similar performance. Therefore, this work aims to answer the following questions:

Q1 – Is classification of residues in images better with a transfer learning model or through a network built and customized from scratch?

Q2 – How close will the developed algorithm be to the accuracy rate of humans?

1.3. Objectives

The key orientation of this research is the development of a proof of concept to help the management of urban waste collection in Lisbon. In order to minimize the environmental impact and improve the management of waste collection in the city of Lisbon, a system of convoluted neural networks is proposed to detect the improper disposal of waste, outside the disposal equipment intended for that purpose, in the Lisbon area.

With the development of this prototype, it is expected that it can perform the following functions:

- Classification of images from different acquisition sources;
- To roughly estimate the amount of improperly deposited waste;
- Identification of trash in the analyzed images;

In addition to these functions presented above, the prototype is intended to be a compromise solution between hit rate and computational complexity.

1.4. Research Methodology

The methodology followed in this work is based on the Design Science Research (DSR) model. This methodology is adequate for solving real problems and is oriented to the creation of artefacts [15]. The DSR model defines a set of essential steps that will lead to the construction of a final artefact, as can be observed in Figure 1.

After the problem identification, the first stage on the iterative process corresponds to the objective's definition, leading to the formulation of research questions that are expected to be answered with the realization of this dissertation. This stage is addressed in sections 1.1, 1.2 and 1.3 of this Dissertation.

The next stage is the design and the development step, where the artefact is defined. In this case the artefact is the garbage detection model based on convolutional neural network. It will be developed following an iterative approach similar to the agile methodology of Software development [4].

Then, the demonstration stage puts into practice the verification of the robustness of the model. It provides details and explanations on how the model is trained to detect and classify the objects through the images. Besides the demonstration of test experiments or simulations, preliminary results are expected to be produced in this stage.

In the evaluation stage, a set of performance metrics enable to draw conclusions about the efficiency of the artefact developed. Allows checking the results obtained between this phase and the demonstration phase. It is also possible to compare the results obtained with related tools.

The final stage consists of the communication of the artefact where its usability and utility are demonstrated through writing the dissertation and an article, which will possibly be published in a scientific journal.

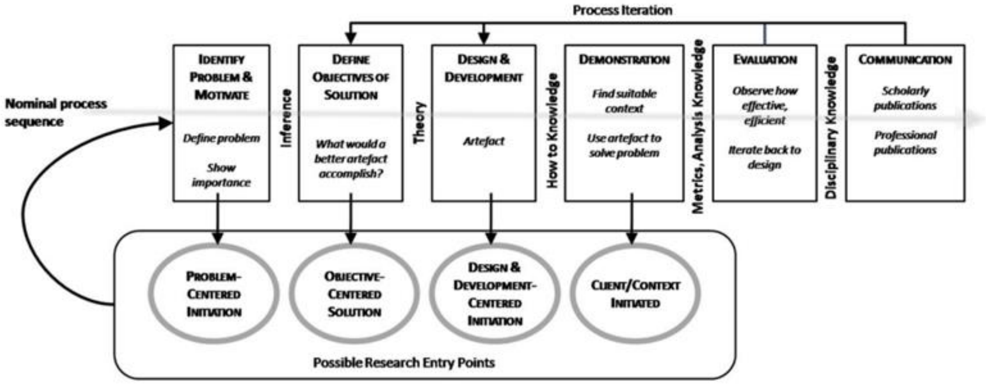

Process Iteration
Nominal process sequence
Identify Problem & Motivate
Define problem
Show importance
Inference
Define Objectives of Solution
What would a better artefact accomplish?
Theory
Design & Development
Artefact
How to Knowledge
Demonstration
Find suitable context
Use artefact to solve problem
Metrics, Analysis Knowledge
Evaluation
Observe how effective, efficient
Iterate back to design
Disciplinary Knowledge
Communication
Scholarly publications
Professional publications
Problem-Centered Initiation
Objective-Centered Solution
Design & Development-Centered Initiation
Client/Context Initiated
Possible Research Entry Points

Figure 1-Design Science Research Methodology (Adapted from Peffers et al.2008)

The final artefact should consist of an automatic classification system capable of recognizing waste outside the waste disposal equipment, using machine learning based in Computer Vision techniques.

On the design and development phase, the tools considered were different automatic learning models that use *Keras*, *Tensorflow* and *OpenCV* free software packages, which provide comparable elements that will allow us to measure the efficiency of the developed system.

Therefore, the points for the development of the solution for this dissertation are presented below:

- Obtaining a robust dataset with as many urban waste images as possible.
- Creation of a deep learning model capable identifying waste in the images it will receive as input, using convolutional neural networks.
- After identifying the waste, its amount is estimated.
- Generation of automatic annotations for the images that are analyzed.

1.5. Structure and organization of the dissertation

The dissertation is organized according to the following chapters:

- The first chapter introduces the subject of study, the motivation, the research questions, the objectives, and the research methodology model used for in the scope of the dissertation.

- An introduction to the main deep learning concepts is presented in Chapter 2, with an emphasis in convolutional neural networks in order to better understand the content of the following chapters. Additionally, this chapter also includes literature review, which depicts a short description of the related work that has already been done on the subject. At the end, a summary of the state-of-the-art research is performed and related with the subject of study.

- In Chapter 3 starts with the description of the proposed system to detect residues outside the designated equipment, through a coherent analysis of the problem to be solved and how the functional prototype built can answer the research questions. The dataset used for system training is also described, from how the data was obtained to the ideal format to perform better.

- Chapter 4 describes the experiments and results comparisons in order to find out which model architecture leads to the best results. Each module of the system architecture is explained in detail. It also explains the training and classification process of the automatic learning system in the context of the problem. At a later stage, the proposed solution is validated with an analysis of the results obtained from the performed tests.

- Finally, the 5th and last chapter draws the main conclusions of this dissertation and suggests topics for future work that aim to provide hints for the expansion and the evolution of the proposed classification system.

Chapter 2 – Concepts and Related Work

The goal of this chapter is to provide a better knowledge of the technologies that were used to create an automated trash detection system based on image classification. It also includes a literature review of some work that is relevant to thesis's principal propose.

2.1 Deep Learning Concepts

2.1.1 Machine learning

One of the fundamental qualities of intelligence is the ability to learn, which is critical for both human cognitive development and AI [12]. In the realm of AI, machine learning (ML) is the science that enables computers to operate without being specifically taught to do so. Algorithms that interactively learn from data are used in machine learning to offer and choose the information needed for the machine to recognize patterns or similarities in data in order to make accurate predictions.

2.1.1.1 Types of Machine learning

According to Shalev-Shwartz and Ben-David [34], machine learning algorithms can be classified in the 3 following categories: Supervised Learning, Unsupervised Learning e Reinforcement Learning. In this dissertation the algorithm used was from the Supervised Learning category.

Supervised learning algorithms are trained on labeled examples, such as an input where the desired output is already known. For example, a garbage image might have data points labeled "trash" or "no_trash". The learning algorithm receives a set of inputs along with the corresponding correct outputs and learns by comparing the actual output with the correct outputs to find errors. It then modifies the model accordingly. Using methods such as classification, regression, and gradient boosting, supervised learning uses patterns to predict label values on additional unlabeled data.

2.1.1.2 Computer Vision

It is a multidisciplinary field that could broadly be called a subfield of artificial intelligence and machine learning, which may involve the use of specialized methods and make use of general learning algorithms [36].

The goal of computer vision is to have a computer do the same as the human's visual system by classifying, detecting objects, and understanding the scene, which humans excel at.

2.1.2 Deep Learning

Deep Learning can be considered a subset of machine learning where ANN architectures include several layers (hence the term "deep") and learn from large amounts of data. The amount of data generated in today's world is enormous – recent estimates place it at about 2.6 quintillion bytes. Since deep learning algorithms require a lot of data to learn from, this growth in data production is one of the reasons why deep learning capabilities have improved in recent years. Deep learning algorithms benefit from today's higher processing capacity as well as the development of AI as a service [36]. Deep learning enables machines to tackle complicated issues even when they are given a large, unstructured, and interconnected dataset.

2.1.2.1 Convolutional Neural Networks

A Convolutional Neural Network (CNN) is an artificial neural network with an architecture that is designed to learn spatial feature hierarchies and typically applied to images. Figure 2 - Typical CNN architectureFigure 2 illustrates a typical CNN architecture.

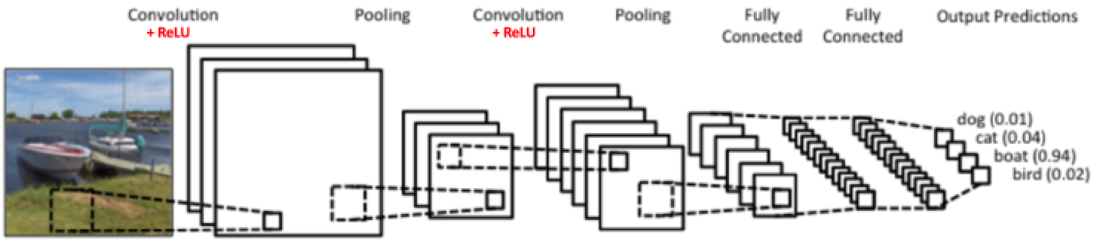


Figure 2 - Typical CNN architecture

A CNN is typically made up of three types of layers:

- **Convolutional Layer** – This type of layer applies convolution operations between a filter kernel and its input data matrix. It is in the training process that the filter coefficients are determined.

- **Activation Layer** – After each convolutional layer, an activation layer (or nonlinear layer) is added with an activation function that assigns nonlinear properties to the input matrix produced by the previous convolutional layer. When developing a CNN, the activation function to be used is usually configurable. The Rectified Linear Unit (ReLU) is a popular activation function that introduces non-linearities while keeping an easier optimization process during the model's training.
- **Pooling Layer** – This type of layers is used to reduce the size of the matrices, simplifying the information in the output of the convolutional layer.
- **Fully Connected Layer** – This layer type is linked to the final judgment on which class the initial image supplied as input belongs to. This is because all of the neurons in this layer are linked to all of the neurons in the previous layer, as is true for all of the layers of regular ANN mentioned above. This layer is represented by a vector with the same number of positions as the layer's neurons. it is in the last fully connected layer that the resulting vector is calculated. It contains a percentage of each position, indicating the probability that the input image belongs to the class represented at that position. The class projected to the given image is the one with the highest percentage. The activation function Softmax or Sigmoid is often used in the last fully connected layer for this value distribution, since it expresses the probability that an image belongs to a specific class, which is a more understandable image concept for network programmers and non-programmers alike.

2.1.2.2 Dropout and Overfitting

When a model learns the information and noise in the training data to the point where it degrades the model's performance on fresh data, this is known as overfitting. This means that the model picks up on noise or random fluctuations in the training data and learns them as features. In short, the network learns specific information from the training set samples and is unable to correctly classify new samples. One way to avoid this very frequent problem in CNN training is using the Dropout. Dropout is the process of "turning off" a randomized set of neurons at the start of each iteration on the training process. The neurons that are turned off during a training iteration do not contribute to the network's training in that iteration. Therefore, every time an image enters the network as input for training, the CNN has a different architecture, but all these architectures share the same weights in the links. Because one neuron cannot rely on the presence of another neuron, which may or may not be turned off, this technique reduces the complex adaptations that neurons create with each other. As a result, each neuron is forced to learn more robust features that will be useful for classifying the image with a different set of neurons than the one required previously. Figure 3 shows a dropout example in a neural network[4].

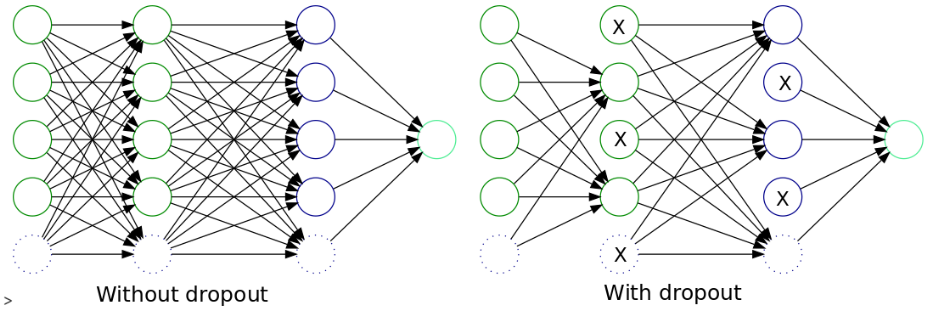


Figure 3-Dropout example, crossed units have been dropped from the network

2.1.2.3 Data Augmentation

Invariance is a property of a convolutional neural network that allows it to classify objects even when placed in different orientations. It would be desirable for CNN's to be invariant to translation, viewpoint, size, or illumination, and can be partially achieved using data augmentation techniques. Because new data is generated based on old ones, this method entails manipulating the data before applying it to the network in order to increase the size of the training set. This strategy also allows the network to rely on the relevant information while preventing an overfit to the secondary details. Cropping, shifting, and rotating are some of the most frequent data augmentation procedures.

2.1.2.4 Transfer Learning

Because of the complexity of the training procedures, training a CNN from scratch is a time consuming and costly process in terms of computational memory. Given that there are several open source platforms with pre-trained CNN architectures, the knowledge acquired by these networks can eventually be re-used in a different problem. This is

[4] https://www.oreilly.com/library/view/machine-learning-for/9781786469878/252b7560-e262-49c4-9c8f-5b78d2eec420.xhtml

typically done by using most weights from a previously trained CNN and modifying the output layers to match the new classification objective. These techniques drive transfer learning, which aims to improve on the traditional machine learning approach by using knowledge from one or more tasks in the original network to access and improve the learning of the new network.

2.2 Literature Review

This section is divided into two parts. The first one contains the relevant review criteria for the literature review. The second part focus in related work, which addresses some applications of the machine learning in waste management. At the end some conclusions are drawn.

2.2.1 Review Criteria

In order to provide a transparent and concise literature review on the recognition of waste outside the disposal equipment using video analytic methods, the process suggested by Briner and Denyer in [30] as well as the characteristics defined in the PRISMA statement in [31] were followed.

The methodological approach, for the review follows a process that involves three stages. In the first stage the goals and needs of the revision are identified, where a proposal for revision is prepared and the criteria are developed to support the revision. Next, comes the second stage, which is geared toward research, quality assessment, data collection and data analysis. Finally, the third stage consists of reporting the results of the review.

A systematic literature search was carried out during the months of November and December 2020, on the subject of recognizing images of waste using Computer Vision techniques. The *Scopus*, *Research Gate*, *Science Direct* and *IEEE* databases were searched in order to find scientific articles in which the terms 'waste', 'computer vision' and 'identification' were searched in all articles. Of the scientific articles found, the content was mostly related to the development of waste classification systems using machine learning. In addition, many Scopus refinement features were used (multiple results refinements in the sense of specific papers, similar articles, related results).

Several articles were excluded because they were focused mainly on the technical aspects of technology and/or the application of machine learning to contexts different from the intended one. Articles related to the Computer Vision area applied to the classification of recyclable waste were also considered. In total, about 50 articles were analyzed of which 21 were considered relevant for this work.

2.2.2 Related Work

This section describes works related to the dissertation theme, which is the use of machine learning techniques for waste recognition and classification, found in the literature.

2.2.2.1 Waste Occupancy in containers

The system proposed in [25] intends to guide garbage trucks to collect garbage only in areas where the container is critically full. The system allows continuous analysis of the data and uses machine learning to estimate the amount of waste produced in the future. These are sent to the cloud in the form of graphs. The alert to the collection teams is performed via email or text message automatically and periodically with the level of waste in the bin. If the threshold established by the authorities is exceeded, the alert is sent. Liu and Jiang [13] proposed an identification method based on computer vision that performs the detection using images, video, or video capture in real time to identify different types of waste containers. Two approaches were used, one using feature detectors/descriptors and the other using convolutional neural networks. The first used a vector of locally aggregated descriptors (VLAD) and the second used you only look once (YOLO), a neural network of convolution. Another study suggests an intelligent IoT waste segregation bin that can classify and categorize the waste that is disposed of inside it, using the KNN algorithm with the help of sensors data stored in Firebase used by the Google Cloud Server for predicting the status of the bin [26].

The work in [2] deals with the development of a model based on DCNN to classify a waste container as full or not full so that real time waste monitoring systems can later be used to process images acquired by cameras installed near the waste bins or smartphones.

Several known DCNN architectures have been used for testing and training for this task, namely ResNet34, ResNet50, Inception-v4 and DarkNet53. Using K-Fold repeated cross-validation, the models were trained and tested, performing the cross. The results showed that the model with the highest accuracy was Inception-v4, with almost perfect results (accuracy = 0.989, recall = 0.987 and ACC = 0.987).

2.2.2.2 Detect Types of garbage with a view to recycling

Some researched articles suggest the implementation of automatic garbage cans that apply computer vision technology for performing an intelligent garbage separation. Valent et al. [16] propose to use the KNN algorithm, to present an intelligent trash bin method that collects, identifies, and automatically disposes of the garbage in the corresponding bin. Omar et al. [23] proposes an intelligent waste separator, called "Trashcan", to replace the recycling bins. Using a KNN algorithm, the device classifies the received waste and position it in different containers. In [32], Salimi et al., present a robotized waste garbage can that uses SVM algorithm to find, define and classify the waste into organic, non-organic and non-waste waste. Considering the problems of traditional industrial waste disposal, such as heavy workload, low efficiency and low safety, a sorting robot was developed in [12]. The proposed system includes intelligent identification, classification, and wireless communication systems. The robot adopts a rectangular coordinate robot structure. After collecting photographic information, the robot can interact with a computer. The SVM algorithm is used for the autonomous sorting and transport of waste information for further classification. To locate and choose the waste, Wang et al. proposed in [17] a solution in which RGB images first are firstly resized to 224x224 pixels, which is the ideal input size for the VGG16 model. Next, a convolutional neural network based on VGG16 architecture was developed using the TensorFlow tool. The model uses the RELU activation function and adds another layer to accelerate the model's convergence speed, while maintaining the accuracy of waste type recognition. Finally, domestic waste is classified into recyclable waste, toxic waste, kitchen waste and other waste. In 2019, [8] for comparative evaluation of algorithms, the different deep learning models were tested in the context of recycling. The models used for the study were: Densenet121, DenseNet169, InceptionResnetV2, MobileNet, Xception, where 'Trashnet'[1] dataset and Adam (stochastic gradient descent replacement optimization algorithm for deep learning models training) and Adadelta (Adagrad's more robust extension that adapts learning rates based on a mobile window of gradient updates, instead of accumulating all past gradients) were used as the optimizers in the neural network models mentioned above. Chu proposed in [22] a multilayer hybrid deep learning system (MHS) to automatically classify waste disposed of by individuals in the urban public area. This system uses a high-resolution camera to capture the image of waste and sensors. The MHS uses a CNN based algorithm to extract image characteristics and a multilayer perceptron (MLP) method to consolidate image characteristics and other characteristic information to classify waste is recyclable. The MHS is trained and validated against manually labelled items, which significantly outperforms a CNN based reference method that relies on image inputs only.

2.2.2.3 Detecting Garbage in the Street

Tiyajamorn et al., in [6] propose a solution to minimize the amount of waste in large dumps, such as Thailand, where the amount of waste is excessive. The solution was to develop a system that can be used in a traditional dump for an automatic waste separation. The system was called 'AlphaTrash' and its function is to recognize the classification of waste types through convolutional neural networks, where the architecture used was Inception-v1.

Melinte et al. in [13] designed a robot capable of collecting waste that is on the ground using a camera to capture the images for further processing. Pre-trained convolutional networks are used, specifically MobileNetV1 with SSD (Single Shots Detector) for classification. In [19], Rahman *et al.* attempted to develop an intelligent vision detection system capable of separating the different grades of paper using first-order characteristics. A statistical approach with intraclass and interclass variation techniques is applied to the feature selection process to build a model database. Finally, the K-Nearest Neighbor (KNN) algorithm is applied for the identification of the paper object class. The remarkable result obtained with the method is the precise identification and dynamic classification of all paper grades using simple image processing techniques.

2.2.2.4 Conveyor Belts Systems

In this type of implementation, the system revolves around a conveyor belt where the waste is collected, such as a "machine vision based robotic garbage sorting system", where the system consists of three main components: a camera, a conveyor belt and an object grabbing manipulator [27]. The camera images are used by the Regions Convolutional Neural Network (RCNN) to locate and classify objects, which defines a subset of regions in the image that may contain an object and then attempts to classify objects in the images. Using a high-speed camera and the extraction of texture features combined with a probabilistic neural network for waste classification, the work in [24] introduces a system for classifying waste in a conveyor belt. Some researchers say that another way of classifying plastic bottles in a conveyor belt system can be done by using a Support Vector Machine (SVM) algorithm for image classification. Baby et al. [17] propose another concept for a device that can use a Hyperplane Nearest Neighbour (HKNN) algorithm to classify solid waste in a conveyor belt and catch the waste with a robotic arm.

2.2.2.5 Input Data

Most of the researched articles focused on the comparative evaluation of machine learning algorithms. It should be noted that in all, the method used to obtain data is quite specific. As is the case [1], in which the proposed project image processing is done from images collected in an acquisition system mounted on a vehicle and only then undergoes classification through deep learning networks to process the location and classify the different types of waste. Another study proposes an application for smartphones called 'SpotGarbage' [28] capable of detecting, classifying and identifying the location of the garbage from images collected by the user using convolutional neural network algorithms (CNN). The model was trained with a dataset called 'Garbage in Images' (GINI).

In some articles the use of optimizations in existing convolutional neural network models to obtain more efficiency in the results was the solution, this is the case the study in [24] where a robot was designed to make automatic classification based on effective image recognition. The convolutional neural network (CNN) model, such as DenseNet121, was used to recognize the types of waste. The reference dataset used was 'TrashNet’, consisting of a total of 2527 images with six different categories of waste was used to evaluate the performance of CNNs. To optimize the model, a genetic algorithm was added, which improved the classification accuracy.

2.2.3 Summary

This literature review can be synthesized by evaluating all the papers that relate to the use of Machine learning algorithms for image classification applied to waste and conclude that the approach with CNN's algorithm custom has a higher predominance with 8 entries, followed by pre-trained model, MobileNet with 4, as can be seen in Figure 4, which presents the distribution of the use of the algorithm within the universe of papers examined. Because some papers analyze several algorithms, the number of entries is greater than the number of papers.

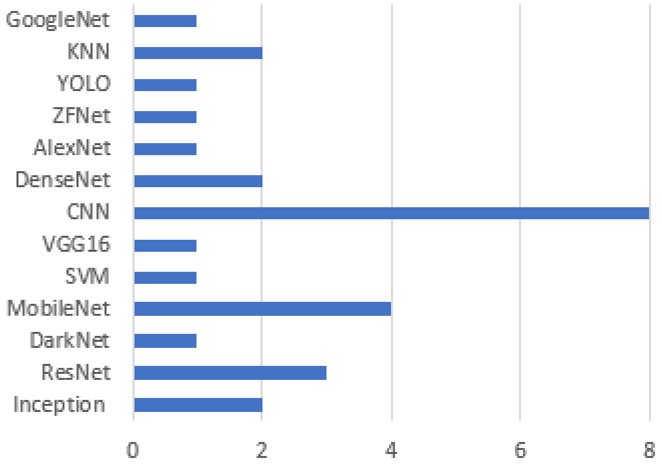


Figure 4 - Algorithm Distribution

Using the information of the title and abstract fields, of the reviewed literature, a visualization of the most frequent terms was built using the applications VOSviewer[5] and Mendeley[6]. The corresponding map can be observed in Figure 5.

An analysis of the most used algorithms based on CNNs, reveals the set of different architectures that can be observed in Figure 5. SVM and DenseNet121 architecture were the most used as solutions for the recognition of waste through images, followed by the architectures Inceptionv1 and MobileNet.

[5] Available: https://www.vosviewer.com/

[6] Available: https://www.mendeley.com/guides/desktop

With the application it was also possible to verify the density with which the terms were used in the articles. The terms 'image', 'classification, 'waste', 'trash', 'computer Vision' and 'identification' were highlighted in the collection of 21 articles studied, as can be observed in Figure 5.

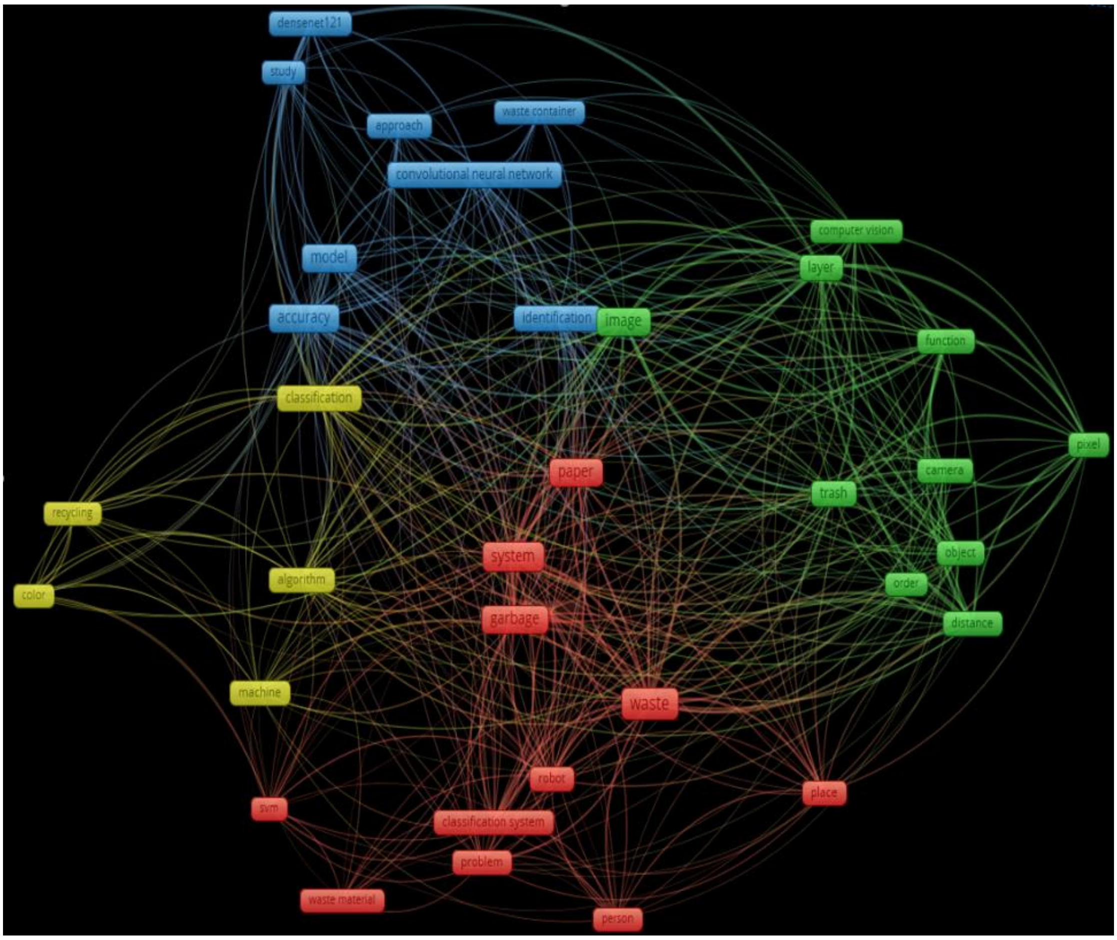


Figure 5 - Network map of collected papers

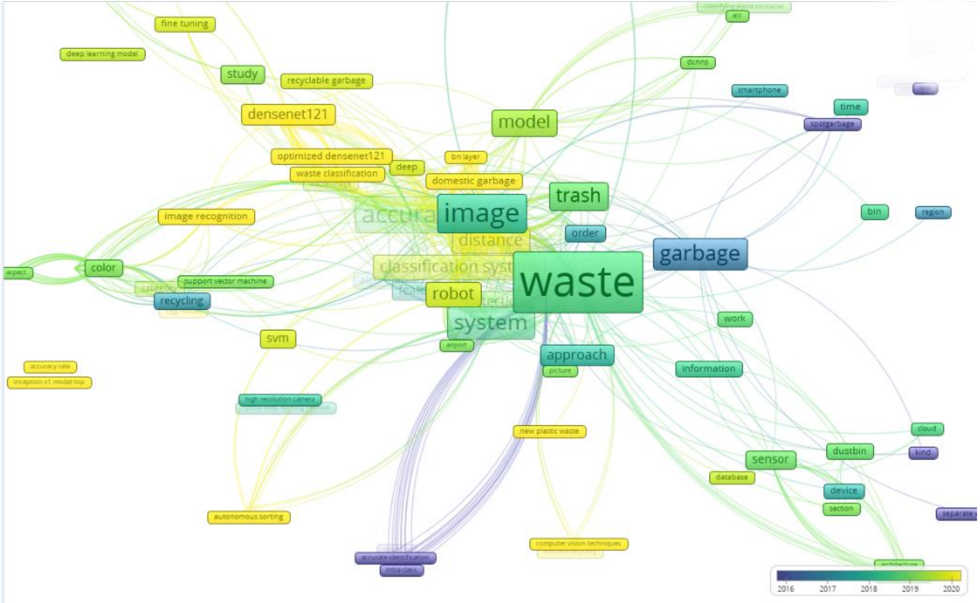


Figure 6 - Map network of the papers content evolution

With this application, it was possible to identify the most mentioned terms in the articles based on their publication years, as shown in Figure 6. With the evolution of technology over the years, automatic methods became more relevant for the realization of projects related to waste recycling. References such as 'robot' or 'autonomous sorting' are represented in yellow and even 'app', 'smartphones', 'high resolution camera' quite present in the articles in the year 2020, as well as more optimized methods at the level of existing architectures in the CNN algorithm. With the technological evolution, it was verified that the articles made in 2019 present developed systems that use more technological resources for the optimization of image recognition in relation to the year 2016.

In short, some conclusions can be drawn regarding the 21 documents that were validated as relevant to the topic. One of the most important conclusions to be highlighted is the fact that some studies present a custom dataset, with varied sizes and image categories, while others use datasets published online (Trashnet and GINI), which makes it difficult to compare with other projects in which the images were taken from scratch (vehicle-mounted systems, photographs, videos, smartphone cameras, etc.), or were not referenced. Regarding the ML algorithms used, it can be concluded that Convolutional Neural Networks and SVM are the most used types of algorithms in this field of study, as shown in Table 1.

Article	CNN Models	Results (%)	Dataset
[6]	GoogleNet (Inception v1)	Precision = 94%	Pictures from Google Images

[9]	ResNet34
	ResNet50
	Inception-v4
	DarkNet53
		Accuracy = 96.8%
		Accuracy = 97.5%
		Accuracy = 98.9%
		Accuracy = 97.2%
			500 images

[11]	MobileNet	Accuracy = 77.3%	Sensor data images
[14]	SVM	Precision = 99%	Trashnet
[17]	VGG16	Accuracy = 75.6%	Communication online about waste info through images
[12]	MobileNetv1	No info available	Images captured by a camera in real time
[1]	CNN	No info available	Images taken from the street and sidewalks
[28]	CNN	Accuracy = 87.7%	Smartphone images
[24]	DenseNet121 optimized	Accuracy = 99.6%	Trashnet

[8]	Densenet121
	DenseNet169
	InceptionResnetV2
	MobileNet
		Accuracy = 89%
		Accuracy= 95%
		Accuracy= 84%
		Accuracy= 95%
			Trashnet

[13]	AlexNet
	ZFNet
	Inception-v1
		Accuracy = 62,5%
		Accuracy = 64%
		Accuracy = 69.8%
			Not mentioned

[9]	MobileNet	Accuracy = 87.2%	2527 images
[16]	YOLO	Accuracy = 87%	Smartphone images
[7]	SVM	Accuracy = 94.7%	Image coveyor belt detector
[19]	KNN	Accuracy = 93%	28,800 papers objects

Table 1 - Synthetized results from the literature

Some results inconsistencies are noticeable. From all algorithms, the CNN algorithm using an Inception-v4 architecture was the one presenting the best results, about 98.9% precision [2]. The literature revision concluded that MobileNet appears to be the ML algorithm with the best compromise between hits and speed. While it is not the most accurate, as shown by the findings of the literature, it was designed to optimize accuracy efficiently when working with optimized methods.

It is also possible to conclude that the work closest to this dissertation are those presented in [6]. This is because it was carried out to identify avoid that the garbage bins did not dye excessive levels of garbage However, there are several points that are not covered:

- The fact that no paper was found that has the same objectives proposed in this project, which is the identification of residues around the containers and the control of the amount of residues produced according to the area. And so, the research criteria had to be broadened.
- The robustness of the dataset and the adaptation of the network architecture used for waste recognition. To obtain good results in terms of hit rate it is necessary that the dataset used to train the network is as robust as possible. Therefore, the more representative the dataset that represents the class to be identified, the more easily the system will be able to detect it. Additionally, the optimization of the network for the desired objective is also important.

With this, most of the authors of the articles reinforce that the success rate of future projects depends on input data, the algorithm to detect and cases of real scenarios in waste recognition systems.

Chapter 3 – Garbage Detection System

This chapter begins with a brief discussion of the computer vision-based system architecture for detecting waste deposited outside the designated equipment. During the literature review in the previous chapter, related work using computer vision techniques for solving waste management problem was identified. However, the requirements presented by the City Hall of Lisbon are focused on different goals than those addressed by the related work, justifying the implementation of a new system.

The necessary requirements for the realization of this system, from the acquisition of images through their processing, will be provided in this chapter. The deep learning algorithm for image classification, as well as the dataset used for testing and training, will also be detailed.

3.1 General Description

Currently, the management of waste collection in the city of Lisbon has margin for improvement on some city areas. The fact that there are schedules for the collections, often leads to be deposited outside the disposal equipment in city areas where the waste production is excessive. In this sense, after debating these issues with city hall representatives, several requirements necessary for the development of this dissertation were defined:

- The management team would share as many images as possible, highlighting waste outside the deposition equipment, along with information regarding each one;
- The system to develop should be based on a supervised learning algorithm with the ability to detect trash outside of the disposal equipment on the images shared by the management team – it should also be able to produce the location of image parts where thrash is present;
- The performance of the classification algorithm system should be evaluated.

Figure 7 illustrates the proposed system for detecting trash outside the disposal equipment:

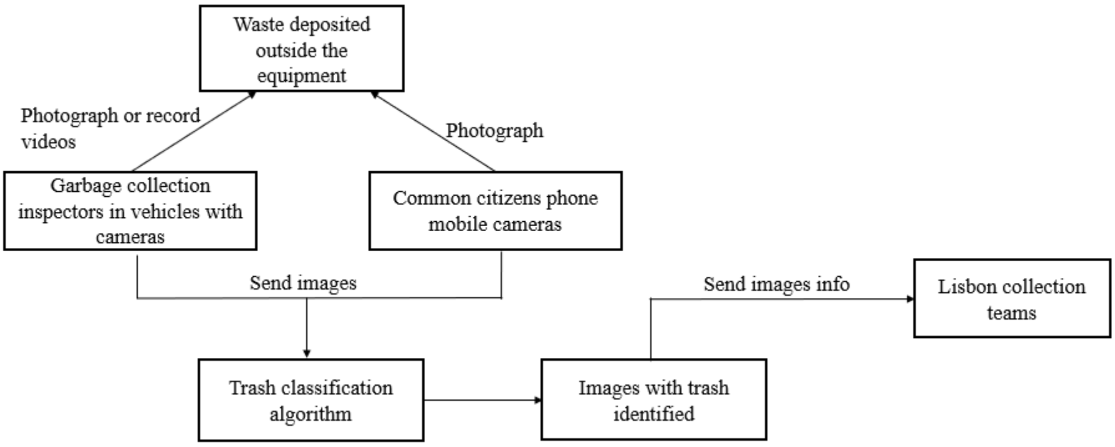

Waste deposited outside the equipment
Photograph or record videos
Photograph
Garbage collection inspectors in vehicles with cameras
Common citizens phone mobile cameras
Send images
Send images info
Lisbon collection teams
Trash classification algorithm
Images with trash identified

Figure 7 – Desired Garbage detection system

The proposed system involves the submission of the images captured by the collection inspectors and those shared by citizens on the app 'A minha Rua' to the classification algorithm. The classification of each image is not done as a whole, but by blocks. Therefore, each block must be classified as trash or not. All blocks classified as trash are identified in the image as a result and in the end this information is sent to the collection teams.

This method would allow a real time quality control of waste collection in areas where waste production is excessive and a better management, in the city of Lisbon. In addition it will be possible to understand where and when to act in comparison to the current collection control procedure.

The first phase of the system implementation is data collection followed by pre-processing. Then this data goes to the waste detection and classification phase, using an algorithm already trained for this purpose.

It is expected that, for a block that does not contain trash, the algorithm associates the class 'no_trash' and for the block identified with trash, the class 'trash'. Figure 8 illustrates this process.

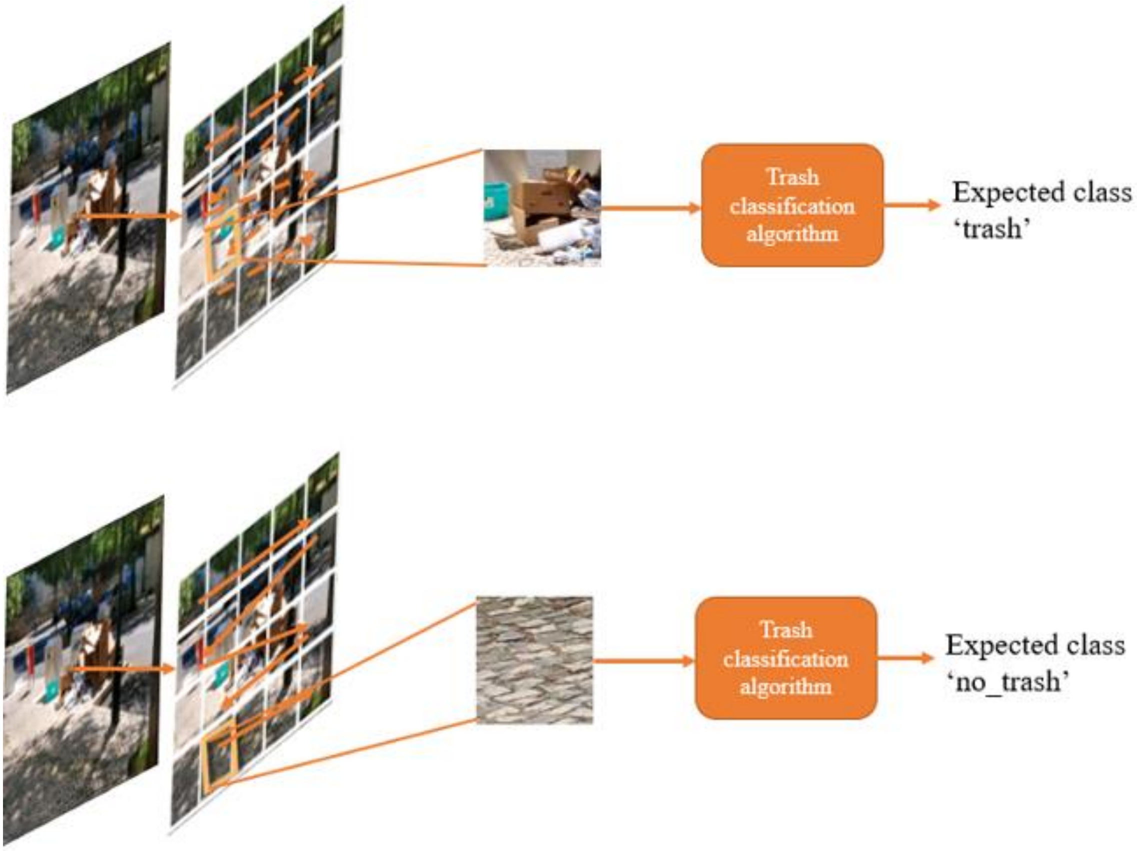

Trash classification algorithm
Expected class 'trash'
Trash classification algorithm
Expected class 'no_trash'

Figure 8 - Algorithm Classification

Finally, the output of the algorithm is an image with the identified residues, as depicted in Figure 9.

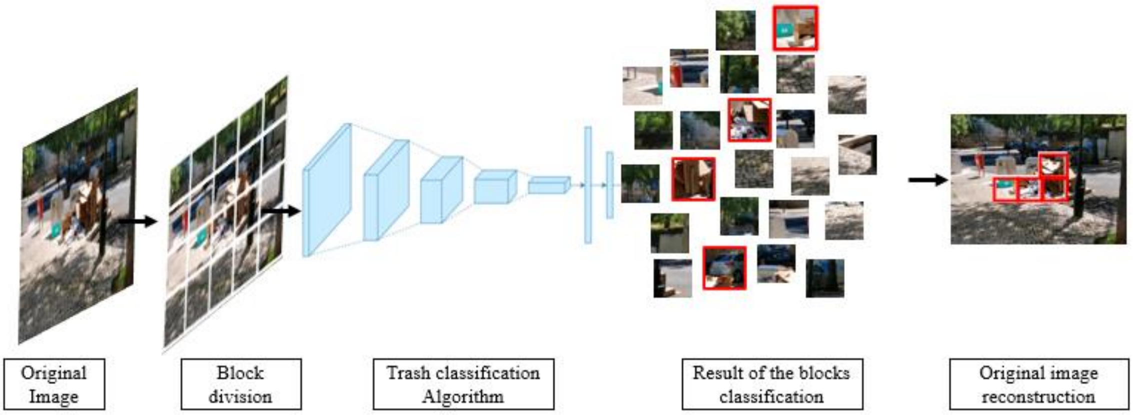

Original Image
Block division
Trash classification Algorithm
Result of the blocks classification
Original image reconstruction

Figure 9 - Expected system behavior

The realization of this work involves collaboration between ISCTE-IUL and the Lisbon City Hall. This dissertation is focused on the development of a binary classification algorithm for the recognition of residues in images. The shared images do not follow any acquisition rules. They were collected without any kind of control, increasing the complexity for achieving the classification algorithm goals.

The following section describes how the classification algorithm works.

3.2 Trash Classification Algorithm

Since during the research of works related to the subject under study no deep learning algorithms were found that dealt directly with the recognition of residues outside the equipment and most of the input images provided are large and obtained without any kind of control, it was decided to start from a simple architecture as a solution between the complexity of implementation and the time required for training and expected results. However, a solution to solve the problem mentioned above would involve research in other domains. As is the case of the dissertation work [41] developed by the student Carolina Gonçalves, in which the main objective of the developed algorithm was to identify invasive species 'Acacia Longifolia'.

The classification algorithm received as input large resolution images previously divided into smaller sub images obtained from a drone. Based on a CNN architecture, the main function of this algorithm is to classify the images into two possible classes 'with' and 'without' invasive species. Since the architecture used in this dissertation work is quite simple and achieved a high value with respect to the classification hit rate, it was decided to follow the same procedure for this dissertation work. It was later modified and adapted according to the results obtained from the experiments performed for the specific case of this dissertation work.

The initial CNN architecture used in this dissertation is represented in Figure 10:

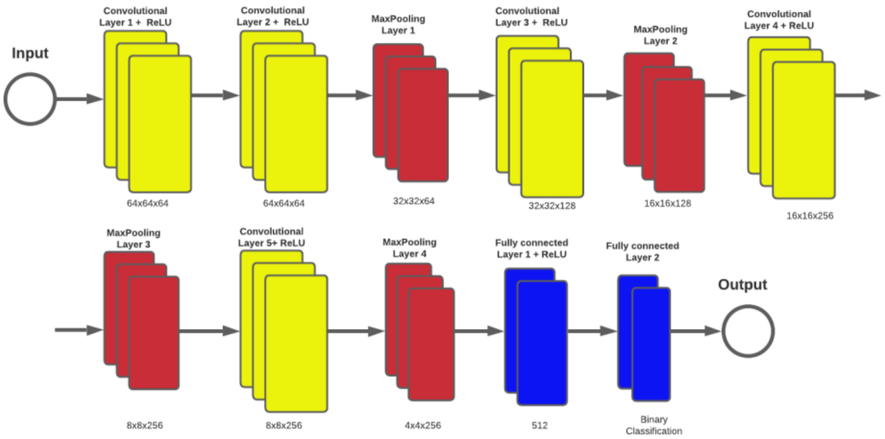


Figure 10 - First architecture setup

This was the first architecture developed. Its inputs are 64 by 64 pixel images. The kernel size for all convolutional layers is 3 except for the second convolutional layer which is 5. The MaxPooling layers with a kernel size of 2. Quite simple architecture containing about 5 convolutional layers interspersed by a Maxpooling layer configured with a stride of 2. The last two layers of the architecture, consisting of fully connected layers, which are basically the 'classical' classification layers based on neural networks. The last layer constitutes the output that allows classifying the ‘no_trash’ or ‘trash’ image. The results obtained in this configuration are explained in detail in the next chapter.

3.3 Data Acquisition for system Training

In this section the creation of the dataset is described, from obtaining the data to how it is treated and placed at the input for the algorithm training.

As mentioned before, the dataset used was provided through a repository of the Lisbon City Hall. Figure 11 demonstrates the flow from data source to algorithm training.

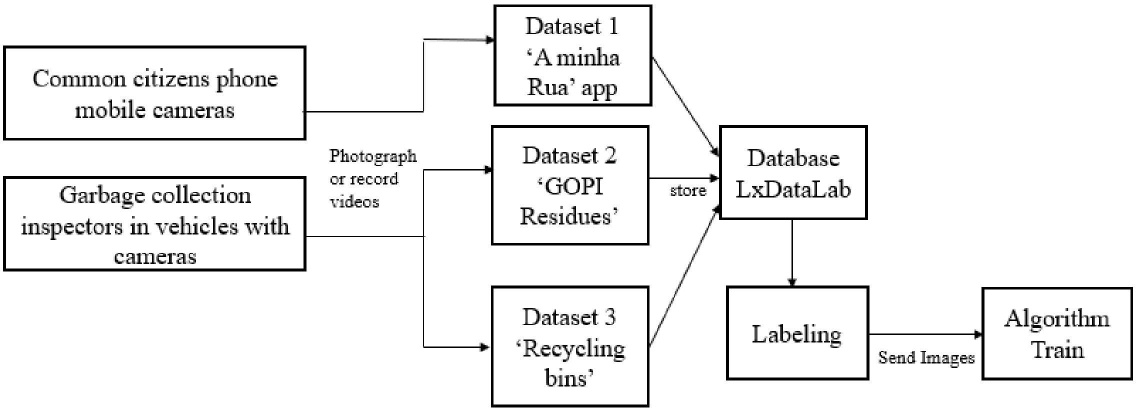

Common citizens phone mobile cameras
Garbage collection inspectors in vehicles with cameras
Photograph or record videos
Dataset 1 'A minha Rua' app
Dataset 2 'GOPI Residues'
Dataset 3 'Recycling bins'
store
Database LxDataLab
Labeling
Send Images
Algorithm Train

Figure 11 - Data acquisition

The process begins with the collection of images by waste collection inspectors or ordinary citizens, of images of waste outside the equipment. These images, previously separated by dataset, constitute the 'LxDataLab' database. Access to the database allows this data to be tagged. Finally the training and testing for the different classification models is performed.

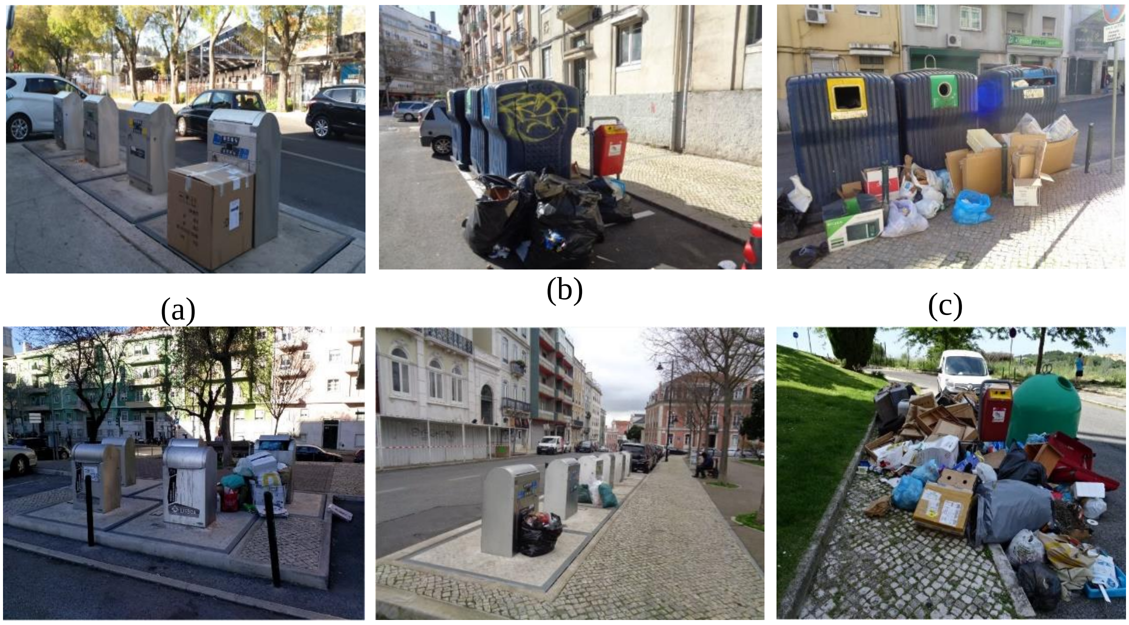
The following two sections explain what was done to generate the data that was used for training the CNN models proposed in the scope of this project.

3.3.1 Dataset

As explained before, the input data – images – is an essential requirement for the development of the garbage detection system based on supervised learning. Data is provided by the Lisbon City Hall, through a private repository called 'LxDataLab', managed by the Lisbon Center for Urban Management and Intelligence. This repository contains data regarding different problems where the use of machine learning may potentially be useful for task automation. For the misplaced garbage detection task, a set of images was collected from several sources. One of such sources is the app 'A minha Rua'[7], where images were acquired by anonymous citizens. Most of these images consist of examples where the waste is deposited outside recycling garbage containers, such as those depicted in Figure 12. In these images, the type of waste that prevails are the

[7] From website: https://naminharualx.cm-lisboa.pt/

common garbage bags, in Figure 12-b), d), e) and f), and cardboard/boxes, in Figure 12-a) and c). LxDataLab'

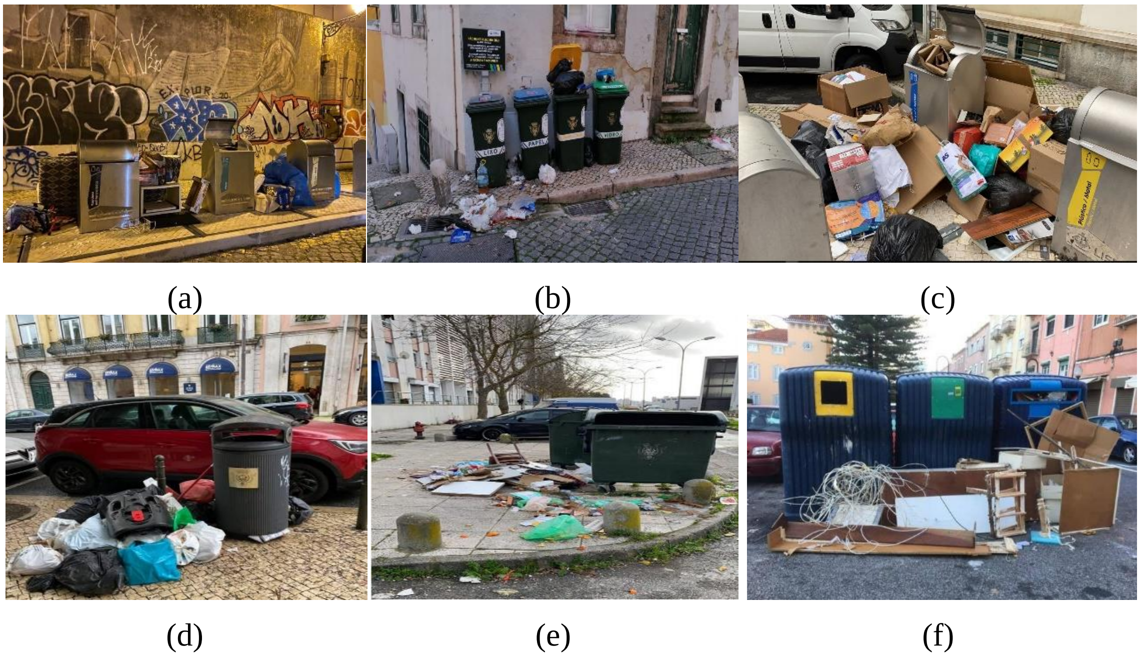


(a) (b) (c)

(d) (e) (f)

Figure 12 - Example of images collected from 'A minha rua' App

Other image sources are smartphone cameras used by the garbage collection inspectors, who snapped images and filmed videos at key locations of Lisbon where trash disposal outside of equipment is prevalent, as shown in Figure 13. There is more variety in the type of waste and disposal equipment in these images, with large residues such as *monos* (labels a and f), biodegradable waste, as in figure (labels b and e), incorrectly deposited waste (label d), and finally places where the equipment cannot support the amount of waste produced (label c) and thus deposited outside the equipment.



(a) (b) (c)

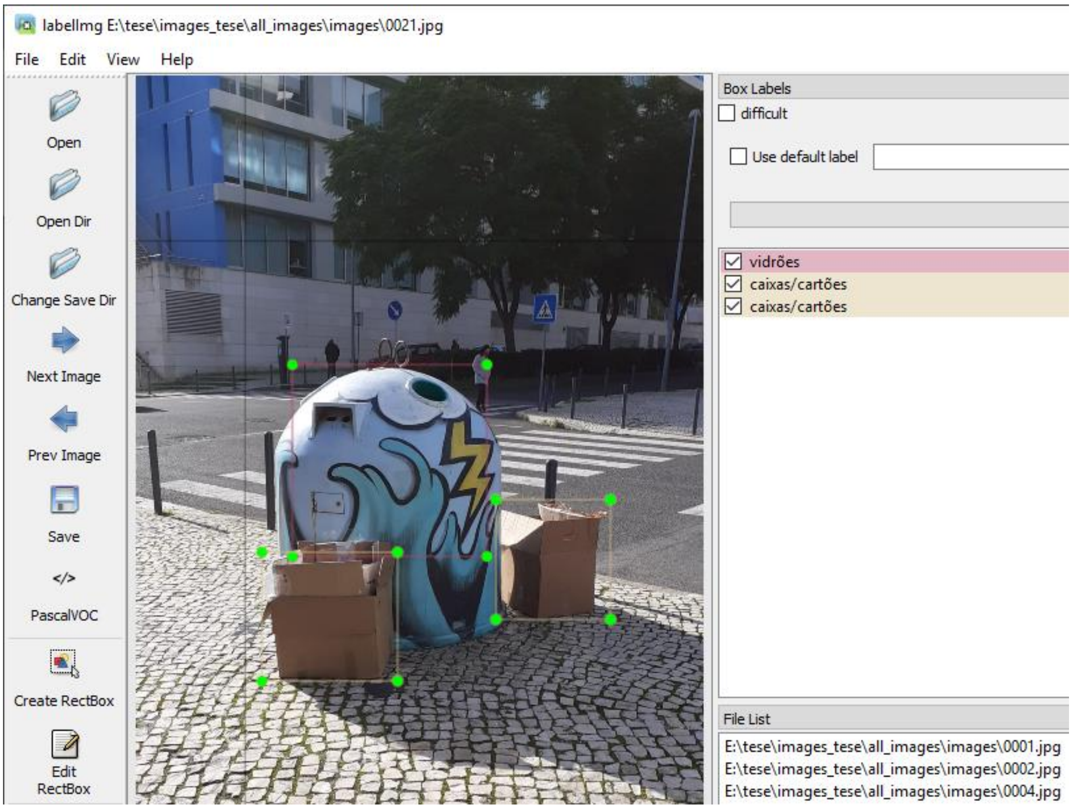
(d) (e) (f)

Figure 13 - Images taken by the collection inspectors

With all these images collected, an unlabeled dataset was built by the LxDataLab team and shared in the scope of this thesis. As previously mentioned, after analyzing the shared images, it was concluded that they were obtained without any control. This led to extensive previous data preparation work.

The first step for the treatment of the images was to count and characterize each one of them. A total of about 1451 images were available. The waste collection inspectors provided 1032 images obtained via smartphone cameras from still positions; 259 were extracted from 5 videos acquired from moving vehicles in the city of Lisbon; and 160 images came from the app 'A minha Rua'.

The second step was the annotation of these images, by characterizing image elements such as: numbering that represents the id of the image, the typology of the containers equipment's, the quantity which represents the residues amount, the type of waste, the location of the residues in the image, resolution and flagged garbage. Some of these characteristics could potentially contribute to a better classification model. The images were initially sorted out according to a new numbering system because their original numbering system was not normalized. The annotations were performed using the labelImg[8] software, a graphic tool that allows an easier image annotation process. With the help of bounding boxes elements were identified as: boxes/cards, loose trash and bags, as shown in Figure 14.



□ difficult

□ Use default label

☑ vidrões
☑ caixas/cartões
☑ caixas/cartões

E:\tese\images_tese\all_images\images\0001.jpg
E:\tese\images_tese\all_images\images\0002.jpg
E:\tese\images_tese\all_images\images\0004.jpg

Figure 14 -Labelling of image 0021.jpg with LabelImg

While performing the labeling process, it was found that there was a noticeable variance in the image resolution, illumination conditions and points of view, which could impose obstacles to the network's learning. The images included in the dataset also contain large portions of background that include elements such as sidewalks, buildings, vegetation, roads, and signs.

Given that a wider content variation on the image dataset could potentially lead to a better network generalization for detecting waste placed outside the equipment, it became critical to collect as many samples as possible. However, the amount of images depicting misplaced garbage was much larger than those without it. In order to overcome this issue, the classification it was decided to classify smaller image patches instead of providing a

[8] Available: https://pypi.org/project/labelImg/

global classification for each image. The original photos were therefore split into smaller 64x64 sub-images, resulting in a greater number of samples that helped in overcoming the mentioned issues. For each sub-image, the number of the corresponding source image and the coordinates of the top-left sub-image pixel were stored in the filenames for future reference. Figure 15 shows examples of sub-images generated from the same source image.

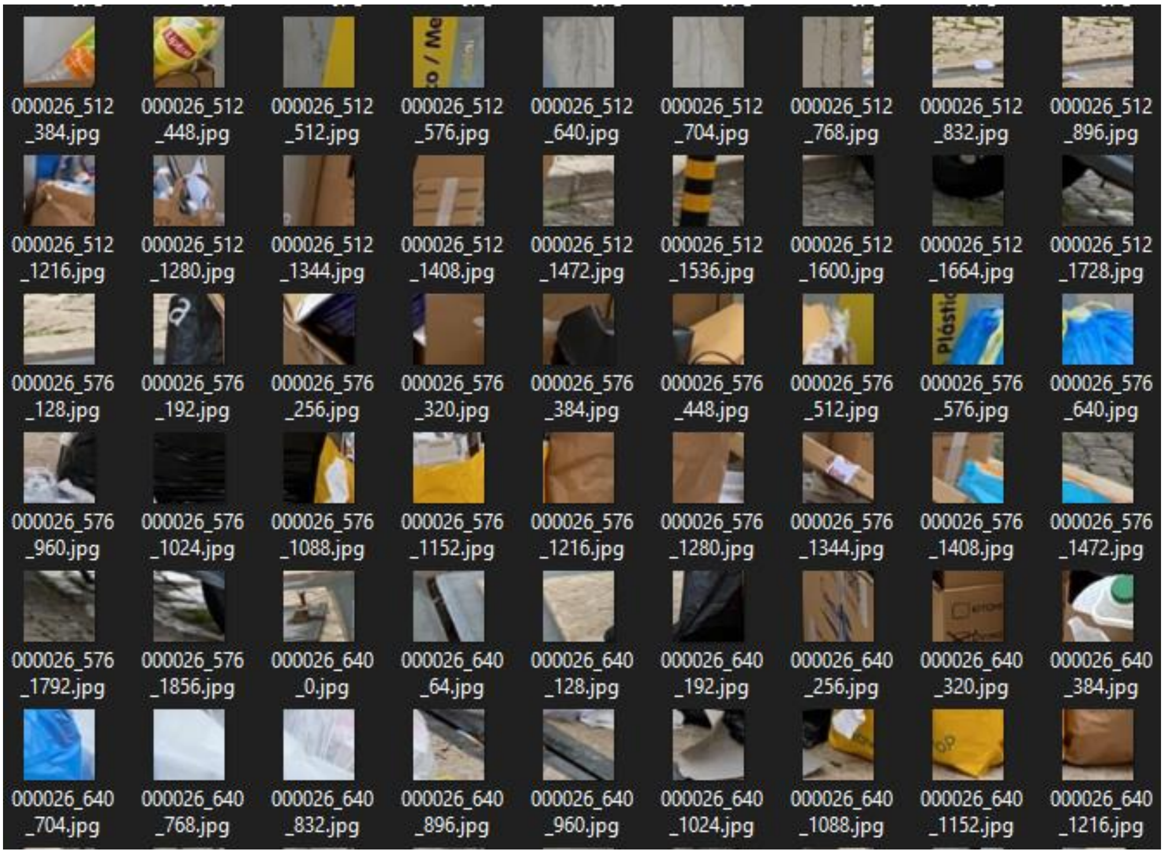


Figure 15 - Creation of sub images from the original images

Each sub-image was then labeled as depicting thrash or no thrash ('trash' and 'no_trash' categories). Each sub-image identified in the LabelImg software keeps the information regarding the position in the original image and the associated label.

For assigning the sub-images to categories the following approach was followed:

If the area saved in the LabelImg software was higher in the 64x64 pixel sub-image resulting from the subdivision of the original image, the sub-image was assigned to the 'trash' class otherwise it was assigned to the 'no_trash' class. The sub-image dataset was organized according to those two categories.

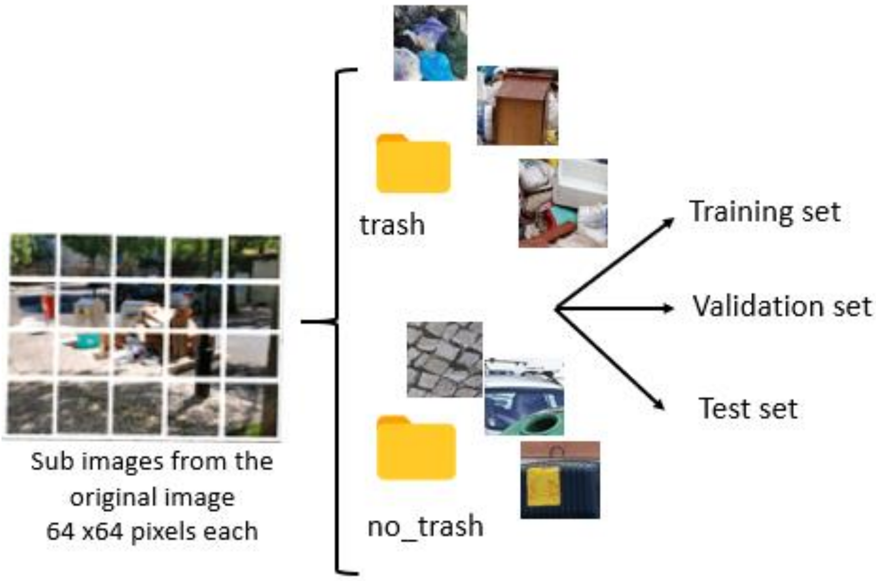


Figure 16 - Class Distribution and Division of the dataset

After setting up the dataset, the sub-images were split into three sets of input data to prepare it for training, validation and testing of the CNN-based machine learning algorithms, as illustrated in Figure 16. The Training data set is made up of data that the model will use to train itself by matching the input to the expected output, which usually is the set with the largest amount of data samples. The validation data set is used for determining how well the training process is performing and can be used for detecting undesirable situations such as overfitting. Finally, the test dataset can be used for evaluating the performance of the model's predictions on new data that was not used during training.

Approximately 19738 samples were used in the final experimental phase, using a split 50/30/20 percent for training, validation, and test, respectively.

There were around 10427 samples in the training set, with 5214 in the 'trash' category and 5213 in the 'no_trash' category, representing 50% of the total. About 5167 samples were identified in the validation set, with 2585 categorized as 'trash' and 2584 as 'no_trash' representing 30% of the input data, and finally about 4144 samples were identified in the test set, with 2136 samples in the 'trash' category and 2008 in the 'no_trash' category, representing 20% of the input data.

3.3.2 Data Organization

Before training the networks for garbage detection, the input images must be prepared to be received by the network. This was done by using data pre-processing and data augmentation techniques.

The ML algorithms used in this work belong to the supervised learning class, and therefore they will be trained and learn from previously labeled examples, where the corresponding output is known in advance. In this sense, an original image can have sub-images labeled as 'trash' or 'no_trash'. The learning algorithm receives a set of inputs along with the corresponding correct outputs and learns by comparing the actual output with the correct outputs to find errors. With this information the model is modified accordingly in order to minimize the error at each iteration.

Thus, it became necessary to define two variables. The features and the labels. The features are the result of the convolutional part of the network entering the fully connected layers. The label is what is intended to be predicted. So in our case for the elaboration of the classifier the features will be the sub-images and the labels will be the categories 'trash' and 'no_trash'.

In order to organize the data structures that allow you to send data to the network for each of the data sets, is created an array of sub-images and their corresponding labels.

This method is applied to the three datasets previously mentioned (training, validation and test), organized into the arrays as described.

3.3.3 Normalization and Data Augmentation

After generating the arrays, the input data is normalized. Normalization consists in defining the range of values in which the data will be transformed to. This process can avoid the saturated values in the activation functions [36].

Thus, the input image data was normalized to the range [0,1] dividing the RGB pixel values by 255.

With the arrays normalized to operate at the correct intervals, data augmentation was used to facilitate network training and to prevent overfitting.

For this, we used the *ImageDataGenerator* class of *keras* available in the *tensorflow* api that allows for real time augmentation of the data. Basically, in each iteration of the training process different versions of images are generated from the original ones. The originals do not enter the training but in the next iteration new "augmented" versions are produced based on them.

The output images after applying this technique have the same dimensions as the input images. The technique was only applied to the training set. Available operations and values of them are shown in Table 2 - Parameter set for data augmentationTable 2.

Operation	**Value**
featurewise_center	False
samplewise_center	False
featurewise_std_normalization	False
samplewise_std_normalization	False
zca_whitening	False
rotation_range	30
zoom_range	0.2
width_shift_range	0.1
height_shift_range	0.1
horizontal_flip	True
vertical_flip	False

Table 2 - Parameter set for data augmentation

At this point the data is ready for the training process.

Chapter 4 – CNN Model Training Experiments

To design and build the training and testing models, the Python programming language and the *tensorflow/keras* library were used. This library provides source code and allows for the rapid creation of code to train ML models. The code developed in the scope of this dissertation was therefore written in Python, running on top of *tensorflow*, all on google *Colab*, a cloud service hosted by google that allows ML and AI research. The *tensorflow/keras* API version used was 2.5.0 and the Python version used was 3.7.11. Since the memory dedicated by google Colab was temporary, the memory dedicated for this project was all allocated to the pc's CPU, so the network training time was quite extensive.

For automatic recognition of residues in the images, two types of convolutional networks were developed and trained: a model built from scratch and a model in which the architecture is already preconfigured and available on *keras*, the MobileNetV2, thus avoiding the creation from scratch and training only the last layer for the intended purpose.

Another relevant variable in the performance of the model is the time it required for training, considering the available resources.

All the experiments were carried out using a HP Elitebook 820 G3, with 32 GB of RAM and an Intel ® CoreTM i7-6500U CPU @ 2.59GHz.

4.1 Hyperparameter Settings

The performance of neural networks with respect to classification is influenced by the values of their hyperparameters. Thus, several experiments were performed, always taking into account the variation of these network hyperparameters, until the final architecture was reached, with a desired hit rate and training execution time. The number of layers, the size of the convolution filters, the number of epochs per training, the probability of random deactivation of neurons in the network, the optimization algorithm and the value of the learning rate were varied.

The Table 3 and Table 4 show the configurations of the tests performed corresponding to the classification in two classes. The configuration and test architecture with the best hit rate was used to build the network that distinguishes, ‘trash’ and ‘no_trash’ classes. Table

3 describes the configurations of the CNN networks developed from scratch, then Table 4 describes the configurations performed on the CNN networks where transfer learning was used.

Experiment	Convolutional Layers	Filters Dimension		Max Pool Layer	Dropout	Ephocs	Optimization Algoritm	Dataset Images	
		1st Layer	Other layers					Train	Validation
1	4	64	128 256 256	4	N/A	200	Adam	2855	2015
2	3	32	32 64	3	0.4	100		10469	5173
3	3	32	32 64	N/A				10427	5167
4	3	32	32 64	N/A				12510	7152
5	5	32	32 32 64 64	3				12510	7152

Experiment	Learning Rate	Activation Function	Training time	accuracy		Loss		Overfitting
				Train	Validation	Train	Validation	
1	10-3	sigmoid	6h	98%	96%	0.02	0.22	No
2	10-6	softmax	4h	76%	68%	0.53	0.60	No
3			3.5h	74%	66%	0.55	0.63	No
4			4.5h	76%	68%	0.51	0.60	Yes
5			2.5h	94%	87%	0.17	0.44	Yes

Table 3 – Custom CNN Model Settings

Experiment	Model	Weights	Dense Layer		Dropout	Ephocs	Optimization Algoritm	Nº Dataset Images	
								Train	Validation
6	MobileNetV2	Imagenet	128	2	0.2	100	Adam	10469	5173
7			1		0.2	75		10427	5167
8			256	2	0.2	100		12510	7152
9			2		0.2	100		10427	5167

Experiment	Learning Rate	Activation Function	Training time	accuracy		Loss		Overfitting
				Train (TL)	Validation(TL)	Train (TL)	Validation(TL)	
2 –6	10exp-5	softmax	1h	89%	80%	0.28	0.42	No
3 -- 9		sigmoid	1.5h	99%	95%	0.04	0.22	Yes
4 -- 7		softmax	1.h	99%	95%	0.05	0.22	Yes
5 -- 8		softmax	1.5h	98%	88%	0.12	0.	Yes

Table 4 - Transfer Learning Model Settings

After the set of experiments performed it was concluded that the model that would be more suitable for recognition of garbage deposited outside the garbage equipment was the model configured in experiment 3 followed by the model with tranfer learning corresponding to experiment 8.

To better explain the experiments performed, the architectures developed for the prototype proposed in this dissertation work will be further detailed, giving special attention to the parameters tunning performed until the final result is obtained.

4.2 Baseline CNN Model

4.2.1 Architecture Description

The initial CNN architecture used in this dissertation is represented in Figure 10, corresponding to the first experience in Table 3.

This was the first architecture developed. Quite simple architecture as explained before in section 3.2. It was necessary to define the optimization and loss functions. The optimization function chosen was Adam [39], because its main function is to measure the mean squared error between each input element, in this case the images, and the categories. This has an adjustable parameter, learning rate, and is mostly used to iteratively update the weights on the training data, in this first configuration the learning rate was set to 10-3. While the loss function used was the binary cross-entropy loss function, this allows to evaluate how good or bad the predicted probabilities are [40]. It should return low values when the neural network is performing good. The activation function used between the convolution and MaxPooling layers was the ReLU. In the last Fully Connected layer, the 'sigmoid' activation function is used since it is the output layer.

At the time this first experiment was done, the dataset did not have as many samples as in its final version. It consisted only of 4870 images. 2855 for the training set and 2015 for the validation set. It was decided to use this CNN architecture repeatedly with variations in the learning rate and the number of epochs with the goal of figuring out which was the best value. Initially the number of epochs configured was 200, however the best model result, i.e. with the lowest validation loss value, may occur before the 200th epoch.

The network is set up for training. In total its training was achieved in about 6h.

4.2.2 Results Evaluation

The results obtained can be observed in Figure 17, where the accuracy and loss metrics were measured. The main purpose of the accuracy metric is to calculate how often the predictions match the labels, while the loss metric is the result of a function that calculates cross-entropy loss between the ground-truth labels and predictions. In the figures shown, the x-axis represents the epochs while the y-axis represents accuracy and loss respectively.

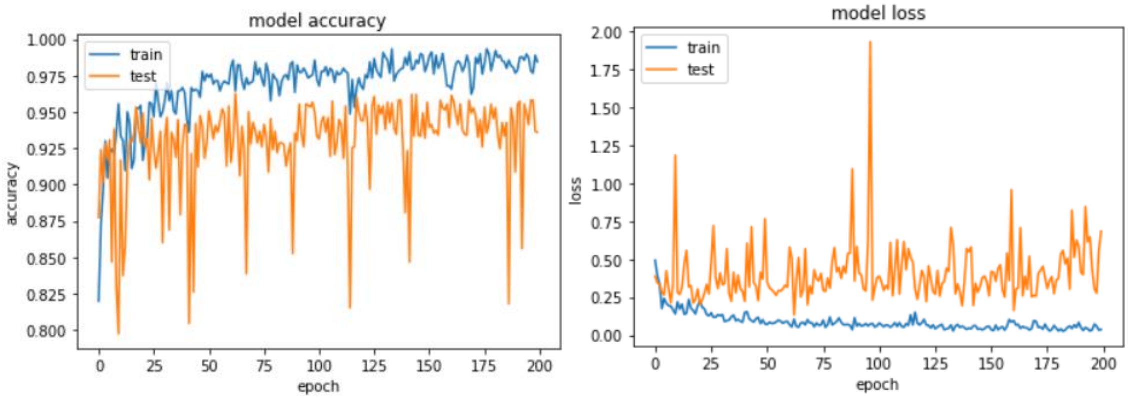


Figure 17 - Model Training results – Experiment 1

Although an initial learning rate of 10-3 was used, a learning rate between 10-7 and 10-3 was also tested in the same architecture, however not showing any difference in the final results in the accuracy and loss charts.

By analyzing the plots, it is possible to conclude that the training values for accuracy (represented in blue) reveal to be different from the validation values (represented in orange), the latter having shown several oscillations throughout the epochs, with values varying between about 80% and 96%. The same happens with the loss metric at the validation value level, with oscillations in the loss values, values varying between 0.22 and 2. Although the net was configured to save the best training results, through the graphs it is possible to conclude that the dataset was unrepresentative for each of the classes under study and too small.

The next sections explain the architecture modifications performed until the desired results were achieved.

4.3 Architecture Modifications Experiments

In order to correct the network training instability problem depicted in the previous section, it was decided to perform changes in the architecture, to increase the number of samples in the dataset and to adjust the training hyperparameters.

4.3.1 Model Variants

Additional tests regarding the architecture of the CNN were also carried out with the goal of understanding the impact of the MaxPooling and Convolutional layers on the performance of the CNN, corresponding to experiments 2 and 3 in Table 3. In both experiments the dataset was increased to 10427 and 10469 samples respectively. A convolutional layer was removed and the filters were decreased. The first convolutional layer was left with 32 filters and the other two with 32 and 64 respectively. In the second experiment one Max Pooling layer was reduced in relation to the first architecture. In the third experiment the MaxPooling layer was removed. A dropout of 40% was added and activation fuction were changed to softmax. Note that the data augmentation technique was initially done both on the training and validation set. For these approaches it was decided to do only on the training set. Also, the number of epochs was decreased to 100 for these experiments. The results were very similar. Huge difference in result graphics in comparison with first Architecture model. However the dataset seems to be not representative as much as pretended, when tested with real images that were not used during model training. To get a model that better classified residues, more images were added to the dataset, and the fourth experiment was performed.

The following figure describes the CNN architecture used, corresponding to the fourth experiment in Table 3 .

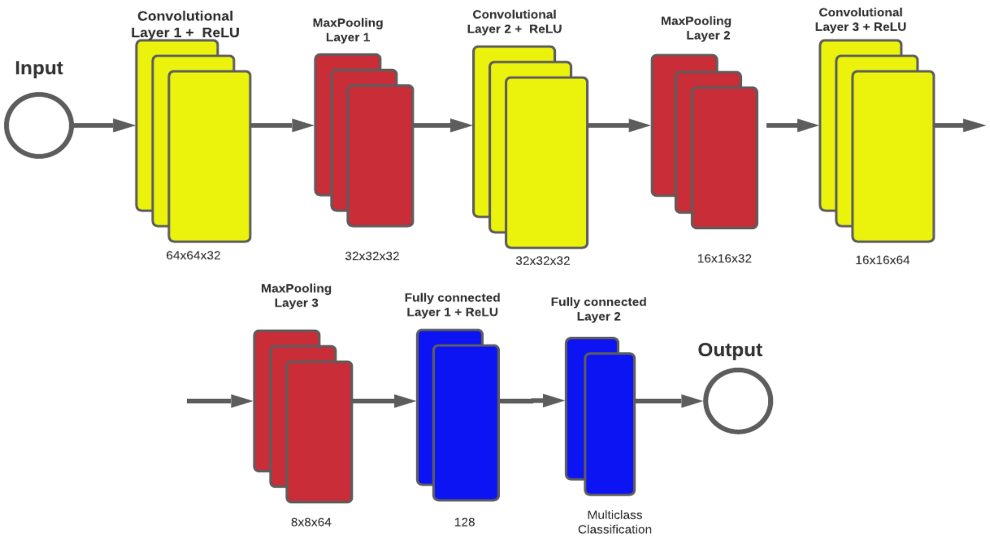

Input
Convolutional Layer 1 + ReLU
64x64x32
MaxPooling Layer 1
32x32x32
Convolutional Layer 2 + ReLU
32x32x32
MaxPooling Layer 2
16x16x32
Convolutional Layer 3 + ReLU
16x16x64
MaxPooling Layer 3
8x8x64
Fully connected Layer 1 + ReLU
128
Fully connected Layer 2
Multiclass Classification
Output

Figure 18 - Architecture setup - Experiment 4

The architecture is even simpler architecture than the initial one. This time with only 3 convolutional layers interspersed with a MaxPooling layer, its inputs are 64 by 64 pixel images. A 40% dropout were maintained for this experiment to minimize the complex adaptations that the neurons have to create between themselves, since they cannot rely on the presence of another neuron as it may or may not be turned off. This forces them to learn more robust characteristics, thus allowing a better classification of the images from a set of neurons [36]. The last two layers are Fully Connected layers, as in the previous architecture.

At the hyperparameter level, as mentioned before, some changes were also performed. The optimization function used remained the Adam but with a learning rate of 10^{-6}. The loss function was changed to the sparse categorical cross-entropy. This loss function performs the same type of loss – categorical crossentropy loss – but works on integer targets instead of one-hot encoded ones. The activation function used between the convolution layers and the first fully connected layer is ReLU with 128 neurons. In the last Fully Connected layer the 'softmax' activation function is used since it is the output layer.

For this architecture, the dataset consisted of 19786 images. 10469 for the training set, 5173 for the validation set and 4144 for the test set.

4.3.2 Results Evaluation

The training of this model was achieved in about 4 hours. The results can be observed in Figure 19, where the accuracy and loss metrics were measured once again.

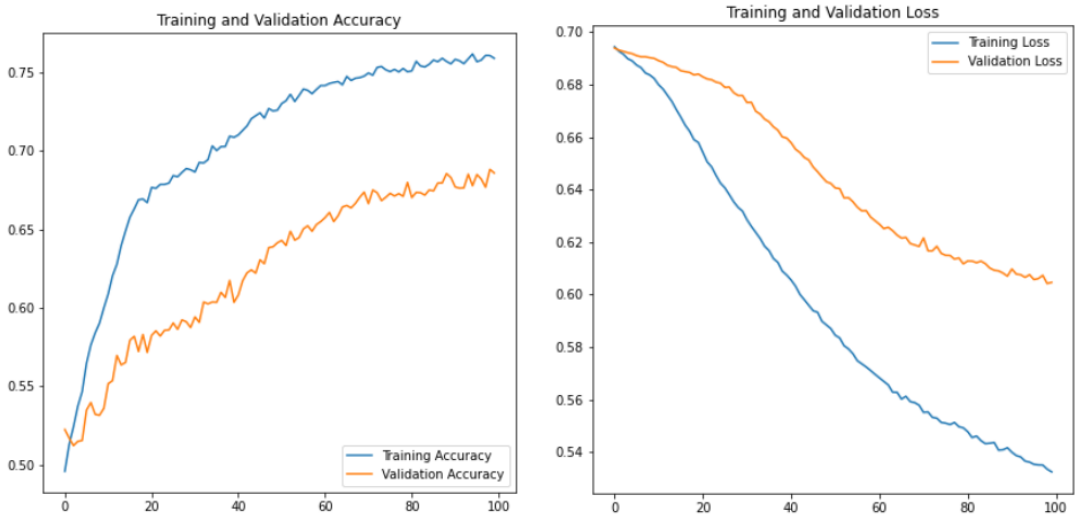


Figure 19 - Model Training results – Experiment 4

It is possible to see a slight difference between the validation values and the training values, with respect to the accuracy metric, however there are still oscillations in the values for both validation and training. Values varying between 50% and 76% for the training set and 53% and 68% for the validation set. Although the accurary values for both the validation set and the training set were increasing, we could see that from epoch 80 on, the values were stabilizing at 70% and 75% for validation and training respectively.

While in the loss metric, there is also a difference from the previous experience, however varying in the range of 0.6 for the validation set and 0.52 for the training set. For both the validation set and the training set the loss values have a tendency to decrease, which indicates that the model could be improved with the addition of more ephocs.

4.3.3 Transfer Learning Experiment

In order to improve the results obtained and have a comparison model, a transfer learning approach was opted. For this, the 'MobileNetV2' architecture available in Tensorflow's Keras was used. This model is implemented to accept images in which the maximum allowed pixels are 224 by 224, which led us to adapt it to accept the size of the images under study (64 by 64 pixels) through the variable 'input_tensor'. The weights that this model uses when imported into tensorflow are those from 'imagenet'. Imagenet is a database of images that contains classifications for these images. Transfer learning is used in this case to speed up the training of the network. Having imported the base model, it then remains to define which layers should be trained, adapting the model to the classification problem under study. A GlobalAveragePooling2D layer is added whose main function is to convert the features into a single vector per image. A 20% dropout was added to the model. The activation function used between the convolution layers and the first fully connected layer is ReLU with 256 neurons. A dense layer with 2 neurons and whose activation function is 'softmax'. We followed the same logic as the previous experiment, at the hyperparameter level: Adam optimization function with learning_rate =10-5; loss function set to 'Sparse CategoricalCrossentropy. The number of epochs as in the previous experiments of 100. The dataset used was the same for each of the sets.

4.3.3.1 Results Evaluation

Training achieved in about 1h, as expected in less time than the previous experiments.

The results for accuracy and loss were measured and can be seen in the following figure:

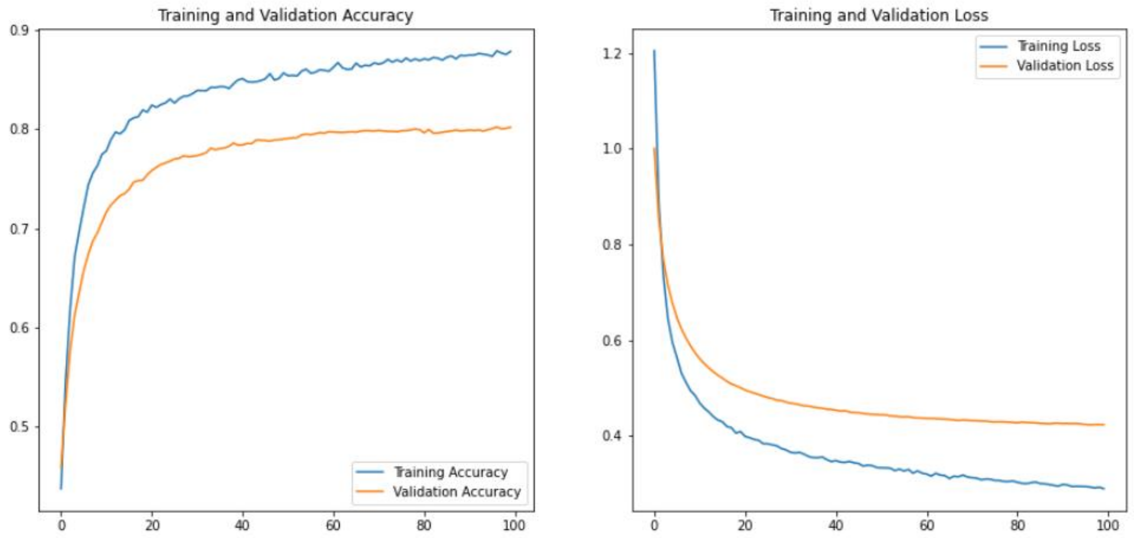
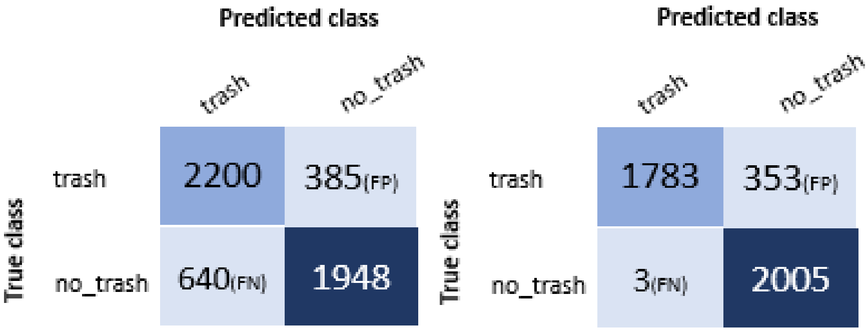


Figure 20 - Model Training results – Experiment 8

In this experiment we have already seen some similarity of results in both the training set and the validation set. Less oscillations in both the accuracy and loss curves are noticeable, when compared with the previous experiments. It is possible to conclude that there was a slight increase in accuracy values in each of the sets. The training set reached 89% and the validation set 80%.

On the loss graph it is possible to conclude that there is a slight decrease in the loss values of both the training set and the validation set. With the training set reaching 0.28 and the validation set 0.42, approximately 0.18 less than the previous experience for the training set and 0.24 for the validation set. It can be seen that the graph of accuracy as of loss from epoch 60 onwards remains constant for the two sets under study. This allows us to conclude that the model is suitable for the recognition of residues in the images.

For further analysis, a confusion matrix was created for each of the datasets under study. This type of table allows to visualize the performance of the model by calculating the number of false positives, false negatives, true positives and true negatives. It is composed of two dimensions: the 'True Class', which is the original classification of the object in the image, and the 'Predicted Class', which is the prediction generated by the model. Each class lies on each of these two dimensions.



Predicted class

True class		trash	no_trash
	trash	2200	385(FP)
	no_trash	640(FN)	1948

Predicted class

True class		trash	no_trash
	trash	1783	353(FP)
	no_trash	3(FN)	2005

Figure 21 - Validation set (left side) and Test set (right side) results confusion matrix

After analyzing the confusion matrices depicted in Figure 21, it was possible to conclude that, for the validation set of the 2585 'trash' images, the model correctly classified 2200 images (true positives) while of the 2588 'no_trash' images 1948 were classified correctly by the model (true negatives). However, 14% of the images that were representing the class 'no_trash' were misclassified (false positives) and 24% of the images that were representing 'no_trash' were misclassified by the model (false negatives). In the test set, whose images were not part of the network training it is possible to verify that of the 2136 images belonging to the class 'trash' 1783 images were correctly classified by the model, while the 2008 images belonging to the class 'garbage_free' were almost all well classified by the model, with only 3 images misclassified.

To further understand the results of the trained model, classification reports were also generated for each of the data sets. In this report, the classification metrics are described for each of the classes under study, which are precision, recall and f1-score.

The classification metric precision was used, since it is usually applied to object detection situations, where we want to evaluate the results of predictions. Precision is the result of the number of correctly predicted images divided by the total number of images belonging to that class [42].

The recall metric, on the other hand, can be interpreted as the measure that calculates the fraction of real positives that are correctly classified by the model [42]. It was found that the trash' class achieved the perfect value reaching 99% while 'trash' did not exceed 83%.

The f1-score combines the two previous measures (precision and recall) in order to obtain a measure that covers the whole range of values of the confusion matrix [42].

The following table describes the values obtained in report classification for the metrics precision, recall and F1-score for the test set. The values were obtained considering that the ‘trash’ class is the positive class and the ‘no_trash’ class is the negative class.

Precision	**Recall**	**F1-Score**
0.83	0.99	0.90

Table 5- Precision, Recall and f1-score values of the previous test set - Experiment 7

The precision tells us how many of the images that were classified by the model (true positives) are correct. From the analysis of Table 5 it is possible to conclude that the class 'trash' is the one that presents the highest value, reaching the perfect value, 99% accuracy followed by class 'no_trash' with 85%. However done for a dataset that was not used during network training, but with similar images. For the validation sets, as they were the same images used when training the network it was found that the accuracy values were lower, as can be seen in Table 6. This means that the model may be overfitting and cannot generalize properly.

Precision	**Recall**	**F1-Score**
0.85	0.77	0.81

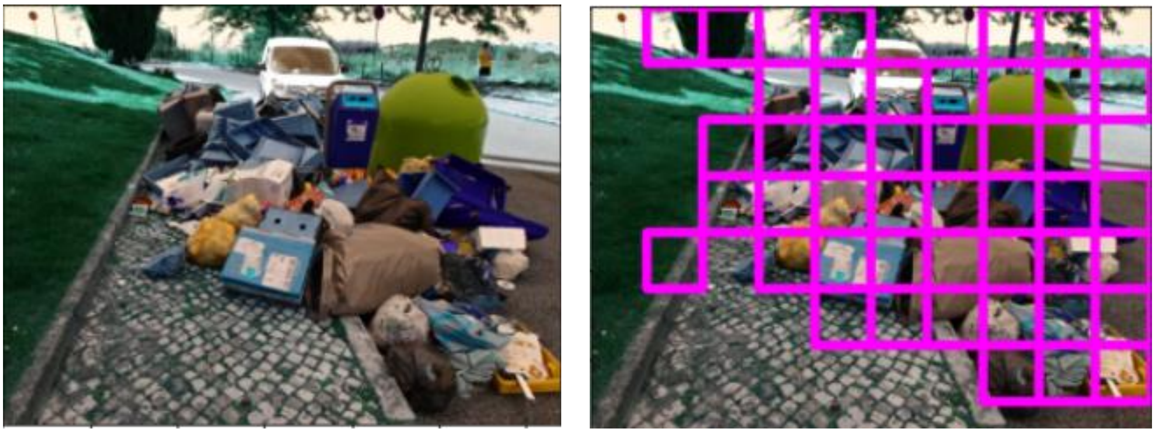
Table 6 - Precision, Recall anf f1-score values for validation set – Experiment 7

4.3.3.2 Full Image Classification Tests

After studying the trained model, where it was found that the values were meeting the expected, it was decided to test the model on real images. As explained initially, after training the model, it is deployed on the proof of concept under study and tested on real scenarios. At the code level the following sequence of steps was followed:

First the original images are placed in a folder (images collected by the camera inspectors without any treatment), then these images are subdivided into images to be received by the already trained model, i.e. into sub-images of 64 by 64 pixels. The third step is to submit these sub-images to the model for classification. Once the entire set of sub-images is classified to the corresponding original image, the system returns this original image again complete and with garbage properly marked. Represented by squares of 64 by 64 pixels. As mentioned earlier, the model goes through 64 by 64 pixels of the original image and classify each of them. Whatever is classified as 'trash' is identified in the original image with a colored square.

The following images represent three images classified by the prototype under study: the left side depicts the original images; the right side depicts the images after the classification.



(a)

(b)

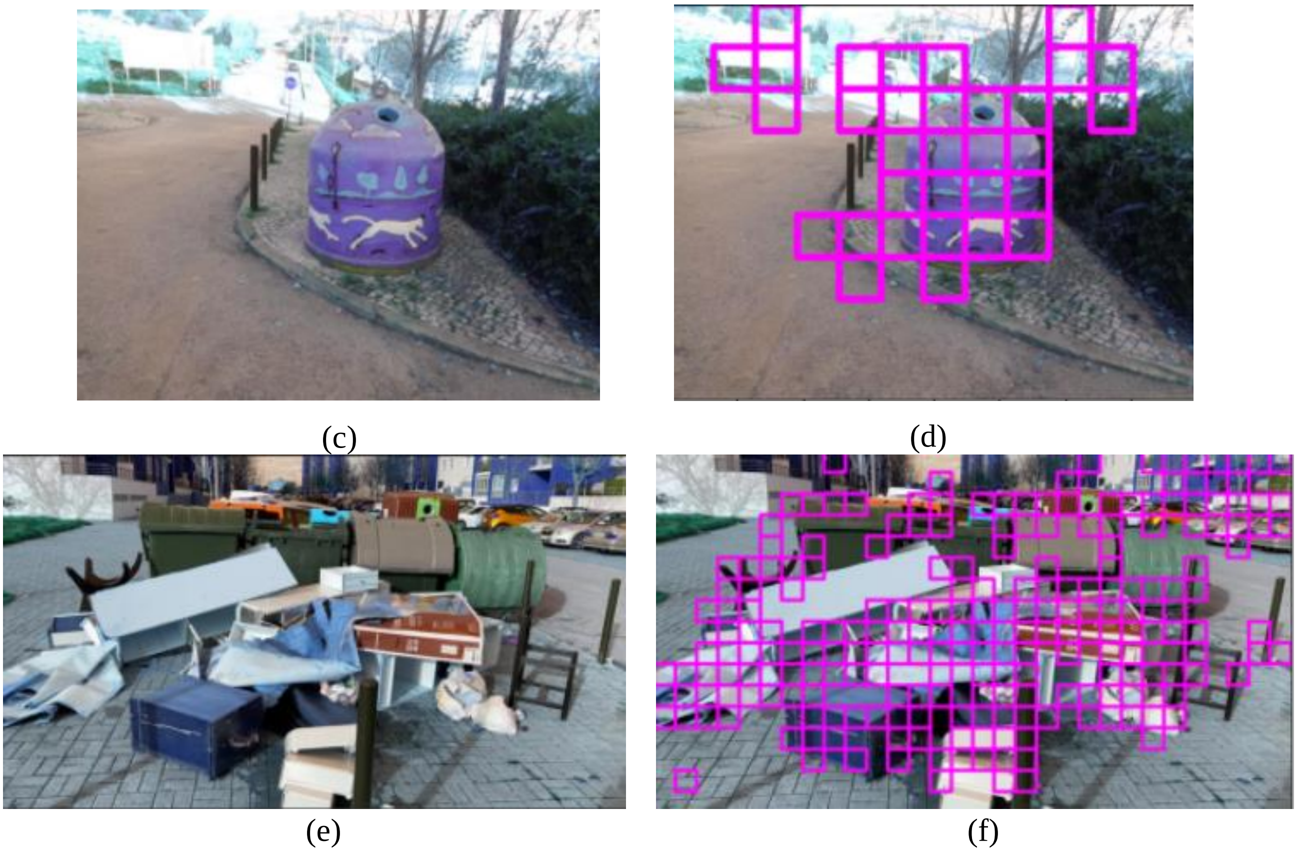


Figure 22- Garbage System residues Identification results - Experiment 7

As can be noticed in the right-side images in Figure 22, many of the sub-images that did not contain garbage were classified as 'trash', i.e. false positives. Considering the confusion matrices depicted in the previous section, this scenario was quite predictable. Figure 22-d) is an example of such situation. It was also possible to verify that the presence of some texture was a sufficient requirement for the sub-image to be considered as 'trash'. Although the accuracy was high, buildings, sidewalks, windows, pillars, cars and the garbage waste containers themselves, were being considered as ‘trash’. It is also worth noting that sub-images that are flat in which a certain color predominates (usually associated with bulky trash or even underground equipment) was also considered as garbage.

The main cause for these results may lie in the dataset used for training, since the images submitted to the model input, come from different sources, many of the images had quite different dimensions with respect to others. For instance, a sub-image of 64 by 64 pixels obtained from an higher resolution image (1980x1750) the model identifies with more difficulty the garbage than an image 580x420 in which the sub-image the model input is more enlarged. This is due to various factors such as texture, brightness, color and zoom. Another possible explanation for why the model is identifying more images with garbage could also be the variety of images split into each of the sets. The dataset may have distinct representativity for the universes of ‘trash’ and ‘no_trash’' sub-images.

4.3.4 Final CNN Model

In order to tackle this problem of false positives, the dataset was updated with a larger variety of possible cases such as and changes in the architecture were performed.

4.3.4.1 New Dataset and Model Changes

The universe of images that characterize the class 'trash' and 'no_trash', for the training and validation sets was enhanced to a more robust dataset consisting of 19662 images (12510 images for training and 7152 images for validation). Where more representative images of the categories under study were added, such as sidewalks, buildings, cars, roads, lampposts, windows and containers. A few structural modifications were also performed in the network by including an additional convolutional layer and a larger number of neurons in the dense layer, as depicted in Figure 23.

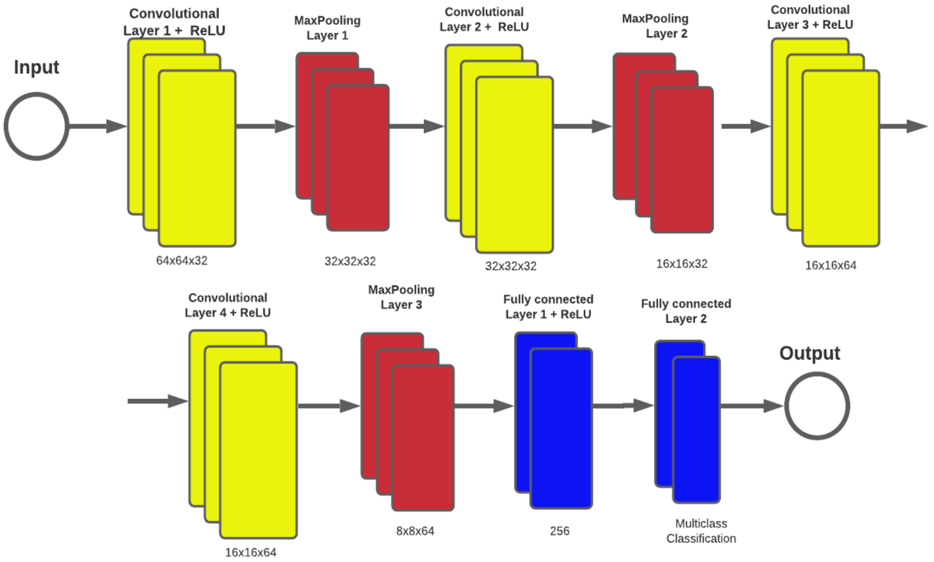

Input
Convolutional Layer 1 + ReLU
64x64x32
MaxPooling Layer 1
32x32x32
Convolutional Layer 2 + ReLU
32x32x32
MaxPooling Layer 2
16x16x32
Convolutional Layer 3 + ReLU
16x16x64
Convolutional Layer 4 + ReLU
16x16x64
MaxPooling Layer 3
8x8x64
Fully connected Layer 1 + ReLU
256
Fully connected Layer 2
Multiclass Classification
Output

Figure 23 - Last Architecture setup

The new network version was trained and analyzed. Not much has changed with respect to the identification of the residues in the original images. With the same dataset transfer learning was used. The only change made in relation to the previous architecture was at the dense layer level to 256 neurons. The metrics accuracy and loss were measured again, as can be seen in Figure 24.

4.3.4.2 Result Evaluation

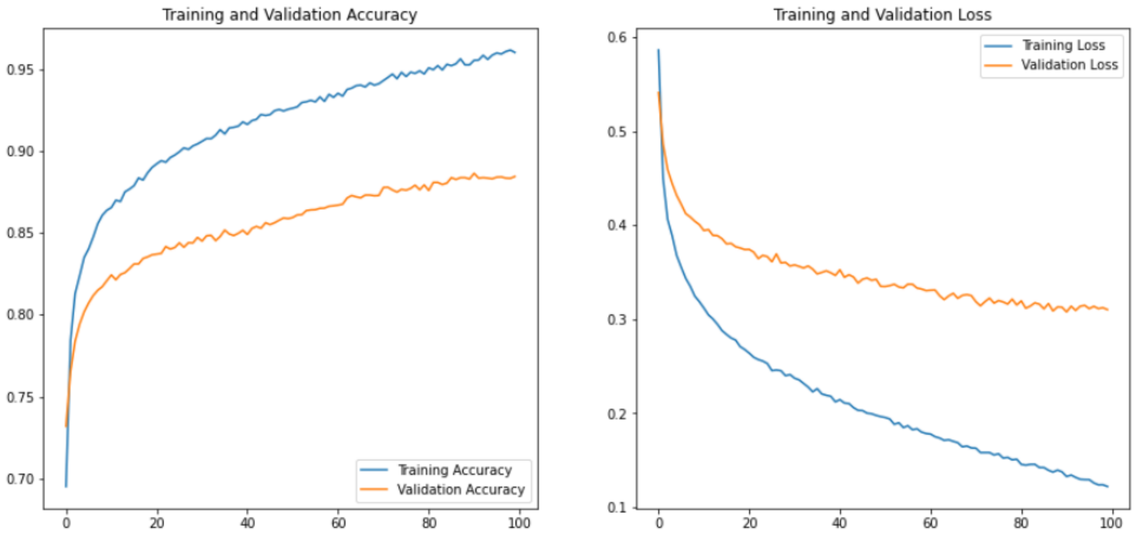
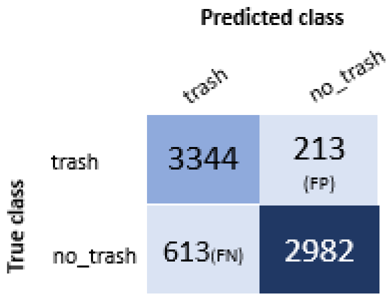


Figure 24 - Accuracy and Loss results - Experiment 8

After analysis, it is possible to conclude that, in relation to the results obtained in previous experiments, and despite some oscillation in the curves, better results were obtained for both accuracy and loss. For the training set there was an exponential increase in accuracy, reaching 98%, while for the validation set there was a slight increase, but from epoch 80 it remains constant, but reaching 90%. While in the loss metric, for the training set there was a sharp and exponential decrease as expected, reaching 0.12, and the validation set varied in 0.30, but with constant values between the 60th and 90th epochs. The conclusion was that the model could be trained further due to the increase in accuracy and the decrease in loss by both the loss variable in the validation set and the accuracy variable.

By generating the classification report and the confusion matrix it was also possible to draw some conclusions. The following table describes the confusion matrix for the validation set.



		Predicted class	
		trash	no_trash
True class	trash	3344	213 (FP)
	no_trash	613(FN)	2982

Figure 25 - Matrix Confusion for validation set results – Experiment 8

It is possible to conclude after analyzing the table in **Error! Reference source not found.**, that with the increase of variety and number of images, that classification done by the model is better than the previous experiment, although there are some false positives that was what was intended to decrease with this experiment. Only 6% were incorrectly classified by the model of the images that existed in the class 'trash'. As for the set of images that represent the 'no_trash' class, 2982 images out of 3595 belonging to the set were correctly classified by the model. This allows us to conclude that it is still not the perfect model, even with the increase in accuracy and decrease in loss, there will be parts of the original image that will be misclassified.

From the classification report the precision, recall and F1-score metrics were also studied and the values obtained are described in Table 7.

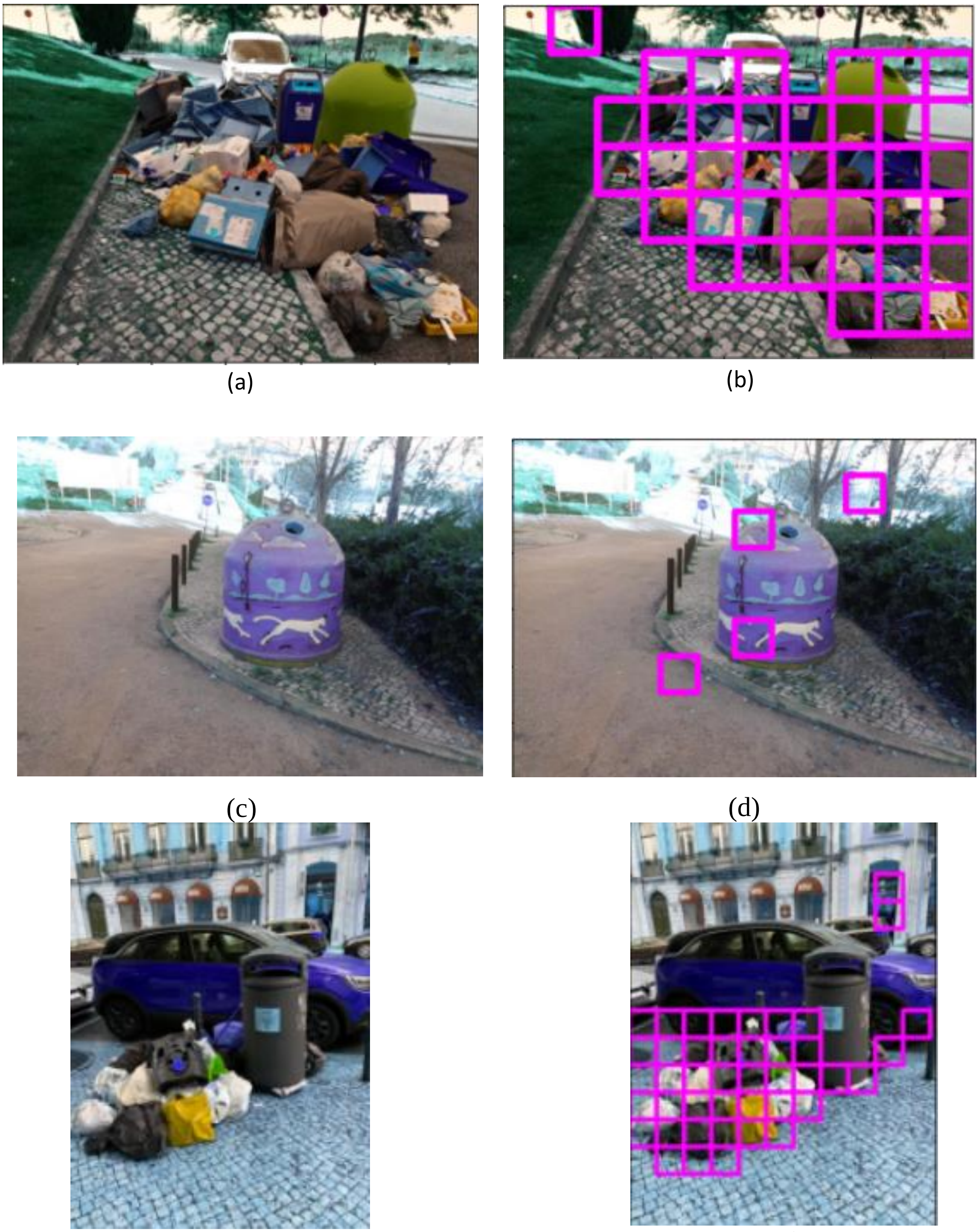
Precision	Recall	F1-Score
0.94	0.85	0.89

Table 7- Precision, Recall and f1-score - Experiment 8

From the analysis of the Table 7, we can mention that for this experiment, the recall reached 94% for the 'trash' class and 83% for the 'no_trash' class and the precision 93% for the 'no_trash' class and 85% for the 'trash' class. The values presented in the table allow us to conclude that the trained model will succeed in identifying and classifying the trash, however, some subimages may be misclassified.

4.3.4.3 Full Image Classification Tests

Testing this model on real cases the following results were obtained:



(a)

(b)

(c)

(d)

(e)

(f)

Figure 26- Garbage System residues Identification results – Experiment 8

As observed in the images on the right side it is possible to verify that, in relation to the previous experiment, although the results were better, the classification is more concentrated in the residues. There are still some sub images classified as 'trash', which was also expected since when obtaining the confusion matrix there was a percentage of false positives, although small, such in the cases of Figure 26-b) and d). It should be noted that even with the increase of images for a better representation of the universe of images corresponding to the classes under study, there were still cases in which windows, tires were identified as 'trash', for instance, Figure 26-f).

4.4 Residues Classification Results

Based on the results obtained, both from accuracy and confusion matrix as well as the classification report, it was concluded that the architecture presented in 4.3.3 section would be the best architecture for recognizing waste outside the designated equipment.

Note that the results obtained in the experiments presented in the configuration tables decreased as the experiments were performed. This is because although the results were getting worse in terms of metrics, the tests on cases were more representative of the intended goal.

It is possible to conclude that the origin of the possible results could be in the dataset used. Considering the origin and the variety of image sizes. It is also important to point out that the results obtained in this dissertation work represent few cases of the possible existing ones. It was found that texture and color factors predominated in the network at the time of classification. From the results obtained in these experiments, it is possible to conclude that many of the images that the camera submits in this prototype created is misclassified, because the system still does not consider a range of factors such as: classification by type of residue, size of the input images, illumination, texture among others. This adapted multiclass classification algorithm acquired knowledge through the data sets created and was optimized considering the problem that was intended to be solved, using Computer Vision techniques.

Chapter 5 – Conclusions and Future Steps

5.1 Conclusions

The goal of this dissertation work was to come up with a possible solution that could contribute for the challenge presented by the City Hall of Lisbon, regarding the detection of waste disposed outside of the designated equipment, using image analysis techniques.

This work does not solve the problem of garbage deposited outside the equipment. However in the future the idea is to recognize the garbage from cameras installed in the collection vehicles of the Lisbon City Hall.

For this, a prototype was created that would be able to detect garbage in images using Computer Vision techniques. The prototype is based on a classification model whose main goal is to detect such waste in the images provided by the team responsible for waste treatment in the city of Lisbon.

The main topics covered in this work were image classification and deep learning using CNN's. Initially, in order to better understand the problem and to check possible approaches to solve it, a literature review in the scope of the application of Computer Vision techniques to Waste Management was performed. This revision, presented in Chapter 2, highlighted related works that were closer to the context of this dissertation. Additionally basic concepts of the machine learning applied to Computer Vision problems, in this chapter as well, with the explanation of the essential factors for the creation and management of a neural network.

In Chapter 3 is presented the system and the necessary requirements, as well as the construction of the dataset and the processing of the input data for training the classification algorithm.

Chapter 4 described the developed prototype, from the processing of the provided images, to the achieved results for the classification models under evaluation. As with any deep learning algorithm, a large amount of data was needed for training and validation, so it was necessary to create and preprocess the data sets.

During this work the dataset at the input of the training model were essential for the obtained results. The dataset created was becoming more robust over time while training the classification algorithm. Always aiming to be the as representative as possible of the universe of classes under study. Which then allowed, through convolutional neural networks, to obtain better results when training the network. The last dataset, which was the last version, both for training and validation, is more realistic and comprehensive. In the experiments with the preliminary datasets, newly created CNN architectures were used followed by transfer learning. At the end of each experiment the trained model was analyzed using the confusion matrix and the classification reports.

The analysis of the results obtained in each of the experiments allowed us to assert if the trained model was meeting the desired requirements. During the first development cycles several changes were performed, both in the architecture and in the dataset. Chapter 4 also presents mechanisms for solving the problems encountered. When obtaining a model that met the intended requirements, we validated the model using real cases.

Regarding the first objective of this work: "being able to classify images from different sources", the results are encouraging, although images required extensive prior treatment before they were presented to the network, since they were obtained without any control, and they were of varying sizes. The process shown enables the creation of an effective training-set from which a reasonably accurate classification rule can be learned.

Estimating the amount of garbage deposited outside the garbage facilities was another proposed goal. It was successful because a relatively low false positive rate was achieved.

Improving the classification architecture and continuously updating the datasets became essential to achieve better results for identifying garbage in the images. The datasets had a preponderant role in this sense. Since the first dataset created led to worse results due to its lack of variety of examples of images with trash and images without trash. Limitations as loss of image quality after resizing the data. However after several experiments, it led to having to build datasets from scratch, more realistic and somehow better represent each of the classes.

These changes in the datasets meant that we had to adjust parameters in the network. The results obtained show that it was indeed possible to adapt a multiclass classification algorithm based on deep learning to this specific problem.

Regarding the questions proposed at the beginning of this dissertation, it is possible to conclude the following:

Q1 – Is classification of residues in images better with a pre-trained CNN or through a network built and customized from scratch?

Yes. With the use of pre-trained networks for recognizing residues in the images, better results were obtained compared to the custom root architectures.

Q2 – How close will the developed algorithm be to the accuracy rate of humans?

Taking into account the experiments performed it is possible to conclude that the configured algorithm in terms of hit rate, is not yet at the level of accuracy of humans, Even though the hit rate for both the validation set and the training set were higher than 80% and 98% respectively, when tested in real scenarios.

It is possible to conclude that the prototype can identify the residues in the images, after analyzing them, however there is still much work to be do to make the system autonomous.

This dissertation allows us to conclude that it is possible, with the help of computer vision techniques, to classify images from different sources and dimensions with the use of the correct architecture. It also allows us to conclude that segmenting the images into smaller images can solve the shortage of data issues.

At this moment it is still far from the desired because, only shared photos and by themselves, do not solve the problem in depth.

Therefore this work, comes to give the first contributions in that direction.

5.2 Future Work

The experiments performed reflected in the results obtained have been positive, although there are still errors. The next step to improve these results should be to improve the data sets by collecting more data.

Also a more accurate labelling differentiating different types of garbage, would be likely to improve results.

A separation of the samples/images by class and resolution is also beneficial, as it is possible that images of various resolutions could compromise the results obtained.

With these dataset upgrades, and retraining the entire CNN we believe results could be improved.

Segmenting the images at the network input may still be a good approach for both training and classifying the images, however experimenting with increasing the size of these input blocks may also be a viable alternative to improve results.

In terms of improvement for the prototype, it should be able to do a more specific waste classification, i.e. classify by type of waste, such as: cardboard/boxes; monos; etc.

Another idea would be to implement a real-time strategy to estimate waste production in areas where dumping is excessive, based on the teams collection history.

Create a waste management app in the areas where there is excessive deposition in the city of Lisbon. This app would be able to update the volume of waste automatically according to a previously configured time interval;

We hope that this work has helped to take another small step towards cleaner and healthier cities.

5.3 Annex A

sensors

MDPI

Article

Identification of the residues deposited outside of the deposition equipment, using video analytics

Soraia Fernandes [1], Tomás Brandão [1], Luís Nunes [1] and João Ferreira [1]*

[1] ISTAR, ISCTE-Instituto Universitário de Lisboa, 1649-026 Lisboa, Portugal; Soraia_Herminia_Fernandes@iscte-iul.pt (S.H.F.); tomas.brandao@iscte-iul.pt(T.B.); luis.nunes@iscte-iul.pt(L.N.); joao.carlos.ferreira@iscte-iul.pt(J.C.F)

* Correspondence: joao.carlos.ferreira@iscte-iul.pt

Abstract: In areas where waste production is excessive, sometimes improper deposition occurs around the equipment of deposition, requiring more effort from the waste collection teams. In this project an image recognition system is proposed for the detection and classification of waste outside the equipment, in Lisbon city, that can be used by collection trucks through installed cameras. To help waste collection planning a repository with several datasets was provided, named 'LxDataLab'. The collected images go through the pre-processing process and finally are submitted to waste detection and classification, through deep learning networks. In this sense, a classification and identification method using neural networks for image analysis is proposed: the first approach consisted in training a deep learning CNN specifically developed to classify residues; in a second approach a CNN was trained using a pre-trained MobileNetV2 model. The training in this approach was faster compared to the previous approach, as were the performance values in detecting the class and the amount of residues in the images. The hit rate for the classification varied between 80% and 98% for the validation set and the training set respectively. After the detection and classification of residues in the images, annotations are generated on the images.

Keywords: Convolutional neural networks; Machine learning; Image processing; Neural Architecture

Citation: Lastname, F.; Lastname, F.; Lastname, F. Title. *Sensors* **2021**, *21*, x. https://doi.org/10.3390/xxxxx

Academic Editor: Firstname Lastname

Received: date
Accepted: date
Published: date

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

1. Introduction

Much of the generated waste, more specifically solid urban waste, is recyclable, which means that all the waste collected goes through the process that transforms used materials into new products. In recent years, the increase of the worldwide population, together with a society that became consumerist, resulted in more production, more consumption and therefore a larger amount of produced waste, which translates into insufficient infrastructures for the collection and treatment of waste, thus causing great harm to the environment.

According to the 2018 Annual Report on Urban Waste , each Portuguese citizen generates an average of about 505kg of waste per year (well above the European average – 476kg/year). The report1 also states that 5.2 million tons of urban waste were collected in Portugal (+21.1kg inhabitant/year of what was generated in 2017) which represents an increase of 4% over the previous year.

Efforts that aim to decrease the statistical values mentioned rely on increasing the percentage of recycling, the economic sustainability of the models that generate waste, and the decrease in the amount of waste that is disposed of in landfills.

Much of the generated waste, more specifically solid urban waste, is recyclable, which means that all the waste collected goes through the process that transforms used materials into new products. Depending on the type of waste, different recycling processes are followed, and therefore, applying methods that allow the correct disposal of waste in the equipment designated for this can bring benefits. The existing techniques that allow the separation of waste, more specifically the selective sorting (garbage recycling containers, glass), the models developed and the set of awareness campaigns in order to facilitate the work of collection, have become essential, but still insufficient to reduce environmental impact.

Excessive garbage generation or insufficient frequency of garbage collection causes citizens to dispose of garbage outside the containers, so automatic detection of such situations can help the collection process.

In order to address this problem, Lisbon City Hall has ongoing strategies such as:

- Installation of underground recycling equipment, hoping to lessen the aesthetic impact that garbage generates in the streets. This type of equipment consists of larger waste containers, when they become full, people often deposit garbage in the vicinity of this equipment.
- Optimization of the waste collection circuits, with methods such as the installation of 1500 sensors in several containers scattered around the city – this measure aims to identify how full the containers are. However, the use of these sensors does not provide information about the accumulation of garbage around the equipment.

The waste collection operation management process is a complex and extensive one. Ensuring it is being done with quality is one of the main targets. However, this process will only succeed if people are motivated to perform the correct recycling of waste. Places where the production of waste is excessive often lead to people placing waste outside of the disposal equipment, because the equipment is already full or by sloppiness, laziness and lack of civility. In other cases, these residues are wrongly placed because people do not know where the correct place is, where they should be placed, in order to be recycled, as is the case of large-sized residues such as furniture and house appliances.

One of the main objectives of computer vision-based systems is to perform tasks that mimic the human visual system, namely the classification and detection of objects, and understanding the context in which the objects are found. However, there is a huge separation between what humans and computers 'see'. For computers to be able to see what humans see, they need an input, which is the form of images. In general, image processing often uses convolution methods to extract the main features. Which means performing multiple matrix multiplications with the matrix that represents the image. With these features it is possible at a later stage to perform detection and classification of objects in an image. In this sense, the waste collection operation could benefit from a Computer Vision based system that analyzes images depicting the vicinity of the waste disposal equipment to determine if there is waste placed outside the containers.

Such system could help the management team to monitor the amount of disposed waste in problematic locations, such as those located in Lisbon.

Currently collection management is performed door to door, and the resource of underground recycling bins has become the first approach to cover the visual pollution. However insufficient to supply the amount of waste produced by local people.

In recent years, deep learning applications based on convolutional neural networks have been applied quite successfully to image classification and object detection problems. However, training a deep learning-based model with sufficient accuracy for a given task implies the use of a robust dataset. In the case of misplaced garbage detection, there is a large variation for both the disposal equipment setup and waste types. This variation will therefore require a larger diversity of images for training the classification systems, in order to achieve an accuracy that will allow to identify the locations requiring an immediate action by the collection team.

Although the recognition of objects by the human brain is usually more accurate, today many computer systems already play the same role as humans, guaranteeing similar performance. Therefore, this work aims to answer the following questions:

RQ1:"Is classification of residues in images better with a transfer learning model or through a network built and customized from scratch?"

RQ2:"How close will the developed algorithm be to the accuracy rate of humans?"

The key orientation of this research is the development of a proof of concept to help the management of urban waste collection in Lisbon. In order to minimize the environmental impact and improve the management of waste collection in the city of Lisbon, a system of convoluted neural networks is proposed to detect the improper disposal of waste, outside the disposal equipment intended for that purpose, in the Lisbon area.

With the development of this prototype, it is expected that it can perform the following functions:

- Classification of images from different acquisition sources;
- To roughly estimate the amount of improperly deposited waste;
- Identification of trash in the analyzed images;

The research is organized as follows: Section 2 provides the conceptual foundation of the research, built from a literature review perspective; Section 3 provide the proposed garbage detection system; Section 4 provides the implementation details to train the system to recognize residues; Section 5 details the application of the system in real images and the results; and Section 6 is the conclusion of the work.

2. Literature Survey

2.1 Review Criteria

In order to provide a transparent and concise literature review on the recognition of waste outside the disposal equipment using video analytic methods, the process suggested by Briner and Denyer in [30] as well as the characteristics defined in the PRISMA statement in [31] were followed.

The methodological approach, for the review follows a process that involves three stages. In the first stage the goals and needs of the revision are identified, where a proposal for revision is prepared and the criteria are developed to support the revision. Next, comes the second stage, which is geared toward research, quality assessment, data collection and data analysis. Finally, the third stage consists of reporting the results of the review.

A systematic literature search was carried out during the months of November and December 2020, on the subject of recognizing images of waste using Computer Vision techniques. The Scopus, Research Gate, Science Direct and IEEE databases were searched in order to find scientific articles in which the terms 'waste', 'computer vision' and 'identification' were searched in all articles. Of the scientific articles found, the content was mostly related to the development of waste classification systems using machine learning. In addition, many Scopus refinement features were used (multiple results refinements in the sense of specific papers, similar articles, related results). Figure 1 represents a flowchart of the plan applied.

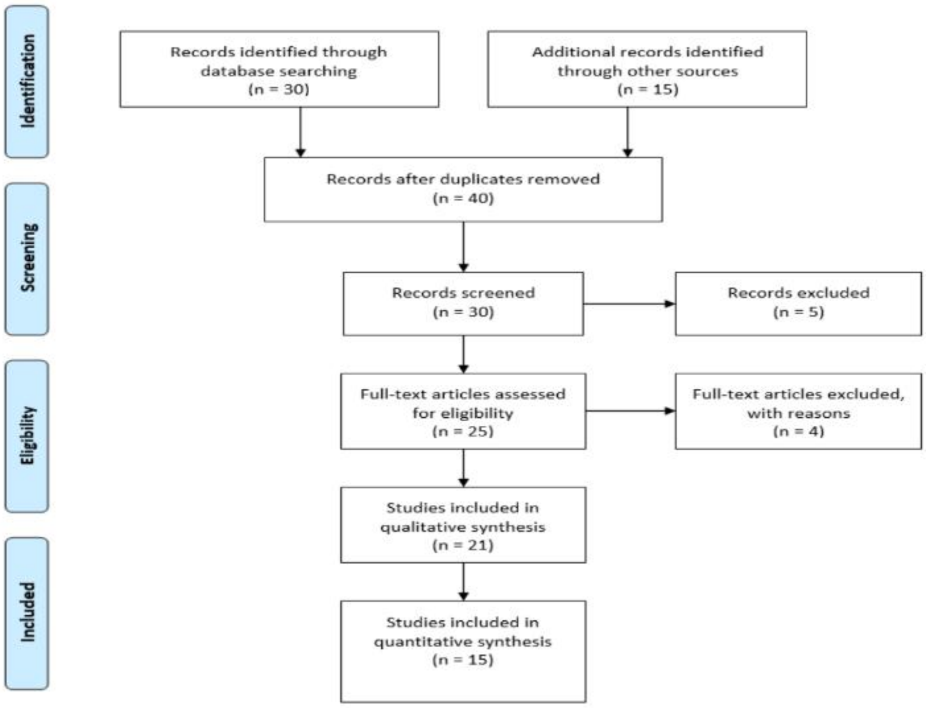

Identification
Screening
Eligibility
Included
Records identified through database searching (n = 30)
Additional records identified through other sources (n = 15)
Records after duplicates removed (n = 40)
Records screened (n = 30)
Records excluded (n = 5)
Full-text articles assessed for eligibility (n = 25)
Full-text articles excluded, with reasons (n = 4)
Studies included in qualitative synthesis (n = 21)
Studies included in quantitative synthesis (n = 15)

Figure 1. Flowchart of the search strategy

Several articles were excluded because they were focused mainly on the technical aspects of technology and/or the application of machine learning to contexts different from the intended one. Articles related to the Computer Vision area applied to the classification of recyclable waste were also considered. In total, about 50 articles were analyzed of which 21 were considered relevant for this work.

2.2 Related Work

The system proposed in [25] intends to guide garbage trucks to collect garbage only in areas where the container is critically full. The system allows continuous analysis of the data and uses machine learning to estimate the amount of waste produced in the future. These are sent to the cloud in the form of graphs. The alert to the collection teams is performed via email or text message automatically and periodically with the level of waste in the bin. If the threshold established by the authorities is exceeded, the alert is sent. Liu and Jiang [13] proposed an identification method based on computer vision that performs the detection using images, video, or video capture in real time to identify different types of waste containers. Two approaches were used, one using feature detectors/descriptors and the other using convolutional neural networks. The first used a vector of locally aggregated descriptors (VLAD) and the second used you only look once (YOLO), a neural network of convolution. Another study suggests an intelligent IoT waste segregation bin that can classify and categorize the waste that is disposed of inside it, using the KNN algorithm with the help of sensors data stored in Firebase used by the Google Cloud Server for predicting the status of the bin [26].

The work in [2] deals with the development of a model based on DCNN to classify a waste container as full or not full so that real time waste monitoring systems can later be used to process images acquired by cameras installed near the waste bins or smartphones.

Several known DCNN architectures have been used for testing and training for this task, namely ResNet34, ResNet50, Inception-v4 and DarkNet53. Using K-Fold repeated cross-validation, the models were trained and tested, performing the cross. The results showed that the model with the highest accuracy was Inception-v4, with almost perfect results (accuracy = 0.989, recall = 0.987 and ACC = 0.987).

Some researched articles suggest the implementation of automatic garbage cans that apply computer vision technology for performing an intelligent garbage separation.

Valent et al. [16] propose to use the KNN algorithm, to present an intelligent trash bin method that collects, identifies, and automatically disposes of the garbage in the corresponding bin. Omar et al. [23] proposes an intelligent waste separator, called "Trashcan", to replace the recycling bins. Using a KNN algorithm, the device classifies the received waste and position it in different containers. In [32], Salimi et al., present a robotized waste garbage can that uses SVM algorithm to find, define and classify the waste into organic, non-organic and non-waste waste. Considering the problems of traditional industrial waste disposal, such as heavy workload, low efficiency and low safety, a sorting robot was developed in [12]. The proposed system includes intelligent identification, classification, and wireless communication systems. The robot adopts a rectangular coordinate robot structure. After collecting photographic information, the robot can interact with a computer. The SVM algorithm is used for the autonomous sorting and transport of waste information for further classification. To locate and choose the waste, Wang et al. proposed in [17] a solution in which RGB images first are firstly resized to 224x224 pixels, which is the ideal input size for the VGG16 model. Next, a convolutional neural network based on VGG16 architecture was developed using the TensorFlow tool. The model uses the RELU activation function and adds another layer to accelerate the model's convergence speed, while maintaining the accuracy of waste type recognition. Finally, domestic waste is classified into recyclable waste, toxic waste, kitchen waste and other waste. In 2019, [8] for comparative evaluation of algorithms, the different deep learning models were tested in the context of recycling. The models used for the study were: Densenet121, DenseNet169, InceptionResnetV2, MobileNet, Xception, where 'Trashnet'[1] dataset and Adam (stochastic gradient descent replacement optimization algorithm for deep learning models training) and Adadelta (Adagrad's more robust extension that adapts learning rates based on a mobile window of gradient updates, instead of accumulating all past gradients) were used as the optimizers in the neural network models mentioned above. Chu proposed in [22] a multilayer hybrid deep learning system (MHS) to automatically classify waste disposed of by individuals in the urban public area. This system uses a high-resolution camera to capture the image of waste and sensors. The MHS uses a CNN based algorithm to extract image characteristics and a multilayer perceptron (MLP) method to consolidate image characteristics and other characteristic information to classify waste is recyclable. The MHS is trained and validated against manually labelled items, which significantly outperforms a CNN based reference method that relies on image inputs only.

Tiyajamorn et al., in [6] propose a solution to minimize the amount of waste in large dumps, such as Thailand, where the amount of waste is excessive. The solution was to develop a system that can be used in a traditional dump for an automatic waste separation. The system was called 'AlphaTrash' and its function is to recognize the classification of waste types through convolutional neural networks, where the architecture used was Inception-v1.

Melinte et al. in [13] designed a robot capable of collecting waste that is on the ground using a camera to capture the images for further processing. Pre-trained convolutional networks are used, specifically MobileNetV1 with SSD (Single Shots Detector) for classification. In [19], Rahman *et al.* attempted to develop an intelligent vision detection system capable of separating the different grades of paper using first-order characteristics. A statistical approach with intraclass and interclass variation techniques is applied to the feature selection process to build a model database. Finally, the K-Nearest Neighbor (KNN) algorithm is applied for the identification of the paper object class. The remarkable result obtained with the method is the precise identification and dynamic classification of all paper grades using simple image processing techniques.

This knowledge can be synthesized by evaluating all the papers that relate to the use of Machine learning algorithms for image classification applied to waste and conclude that the CNN’s algorithm custom has a higher predominance with 8 entries, followed by MobileNet with 4, as can be seen in Figure 2, which presents the distribution of the use of the algorithm within the universe of papers examined. Because some papers analyze several algorithms, the number of entries is greater than the number of papers.

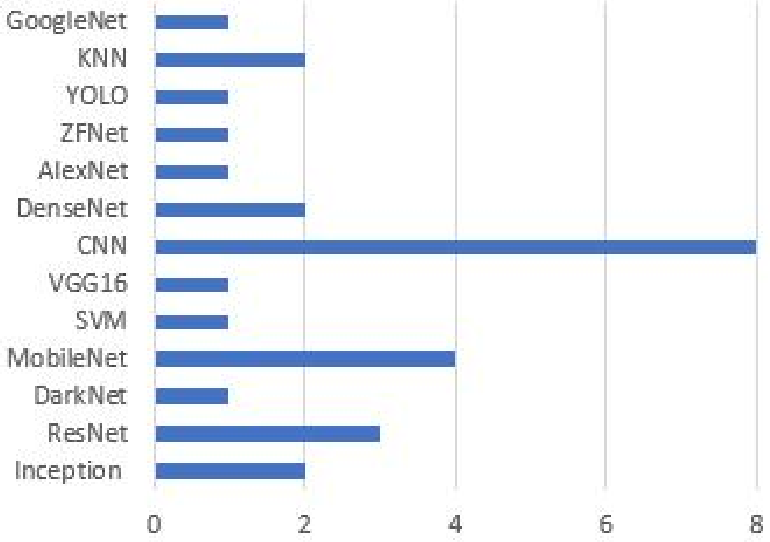


Figure 2. Algorithm Distribution

Using the information of the title and abstract fields, of the reviewed literature, a visualization of the most frequent terms was built using the applications VOSviewer and Mendeley. The corresponding map can be observed in Figure 3.

An analysis of the most used algorithms based on CNNs, reveals the set of different architectures that can be observed in Figure 2. SVM and DenseNet121 architecture were the most used as solutions for the recognition of waste through images, followed by the architectures Inceptionv1 and MobileNet.

With the VOSviewer application, it was also possible to verify the density with which the terms were used in the articles. The terms 'image', 'classification, 'waste', 'trash', 'computer Vision' and 'identification' were highlighted in the collection of 21 articles studied, as can be observed in Figure 3.

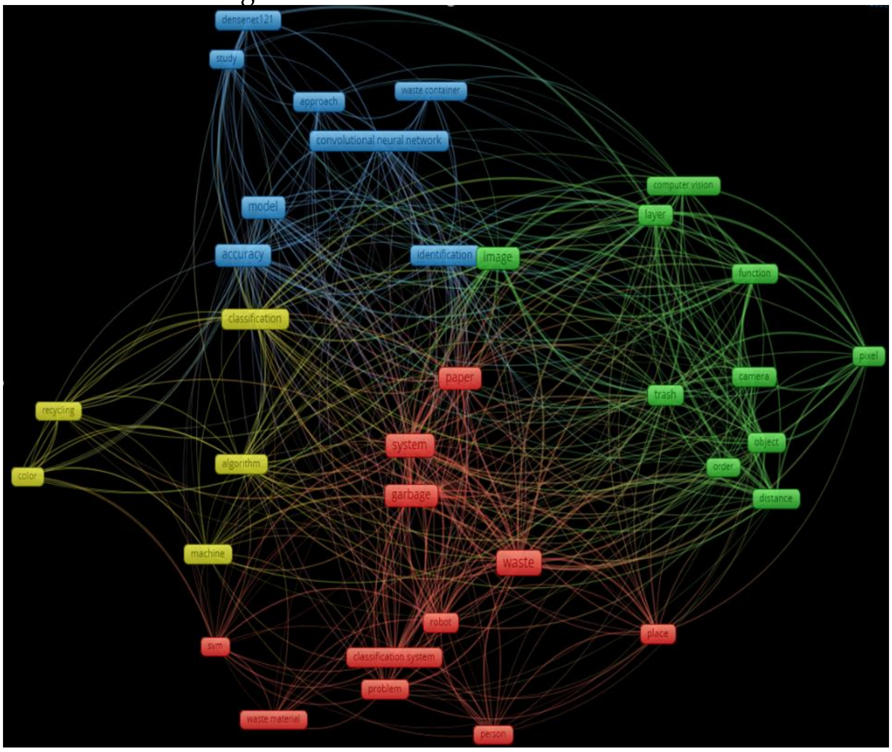


Figure 3. Network map of collected papers

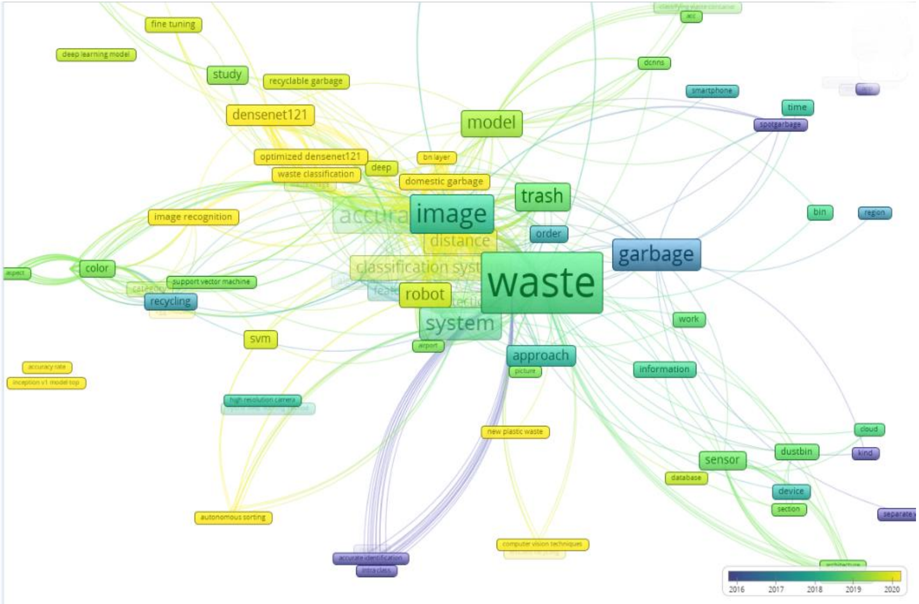


Figure 4. Map network of the papers content evolution

With this application, it was possible to identify the most mentioned terms in the articles based on their publication years, as shown in Figure 4. With the evolution of technology over the years, automatic methods became more relevant for the realization of projects related to waste recycling. References such as 'robot' or 'autonomous sorting' are represented in yellow and even 'app', 'smartphones', 'high resolution camera' quite present in the articles in the year 2020, as well as more optimized methods at the level of existing architectures in the CNN algorithm. With the technological evolution, it was verified that the articles made in 2019 present developed systems that use more technological resources for the optimization of image recognition in relation to the year 2016.

In short, some conclusions can be drawn regarding the 21 documents that were validated as relevant to the topic. One of the most important conclusions to be highlighted is the fact that some studies present a custom dataset, with varied sizes and image categories, while others use datasets published online (Trashnet and GINI), which makes it difficult to compare with other projects in which the images were taken from scratch (vehicle-mounted systems, photographs, videos, smartphone cameras, etc.), or were not referenced. Regarding the ML algorithms used, it can be concluded that Convolutional Neural Networks and SVM are the most used types of algorithms in this field of study, as shown in Figure 2.

Some results inconsistencies are noticeable. For instance, the accuracy varies significantly across different works that used the same algorithms. From all algorithms, the CNN algorithm using an Inception-v4 architecture was the one presenting the best results, about 98.9% precision [2]. The literature revision concluded that MobileNet appears to be the ML algorithm with the best compromise between hits and speed. While it is not the most accurate, as shown by the findings of the literature, it was designed to optimize accuracy efficiently when working with optimized methods.

It is also possible to conclude that the work closest to this project are those presented in [6]. This is because it was carried out to identify avoid that the garbage bins did not dye excessive levels of garbage. However, there are several points that are not covered:

The fact that no paper was found that has the same objectives proposed in this project, which is the identification of residues around the containers and the control of the amount of residues produced according to the area. And so, the research criteria had to be broadened.

The robustness of the dataset and the adaptation of the network architecture used for waste recognition. To obtain good results in terms of hit rate it is necessary that the dataset used to train the network is as robust as possible. Therefore, the more representative the dataset that represents the class to be identified, the more easily the system will be able to detect it. Additionally, the optimization of the network for the desired objective is also important.

With this, most of the authors of the articles reinforce that the success rate of future projects depends on input data the algorithm to detect and cases of real scenarios in waste recognition systems.

3. Garbage Detection System

This section begins with a brief discussion of the computer vision based system architecture for detecting waste deposited outside the designated equipment. During the literature review, related work using computer vision techniques for solving waste management problem was identified. However, the requirements presented by the City Hall of Lisbon are focused on different goals than those addressed by the related work, justifying the implementation of a new system.

The necessary requirements for the realization of this system, from the acquisition of images through their processing, will be provided in this section. The deep learning algorithm for image classification, as well as the dataset used for testing and training, will also be detailed.

3.1. General Description

Currently, the management of waste collection in the city of Lisbon has margin for improvement on some city areas. The fact that there are schedules for the collections, often leads to be deposited outside the disposal equipment in city areas where the waste production is excessive.

In this sense, after debating these issues with city hall representatives, several requirements necessary for the development of this project were defined:

- The management team would share as many images as possible, highlighting waste outside the deposition equipment, along with information regarding each one;
- The system to develop should be based on a supervised learning algorithm with the ability to detect trash outside of the disposal equipment on the images shared by the management team – it should also be able to produce the location of image parts where thrash is present;
- The performance of the classification algorithm system should be evaluated.

Figure 5 illustrates the proposed system for detecting trash outside the disposal equipment:

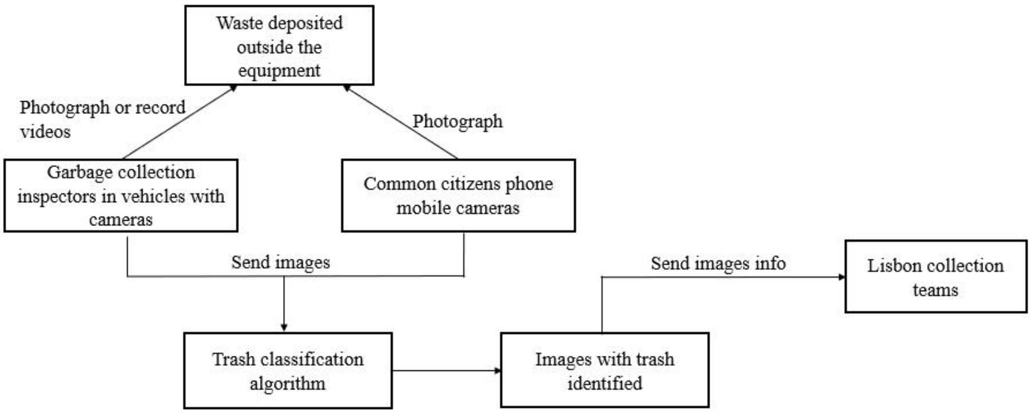

Waste deposited outside the equipment
Photograph or record videos
Photograph
Garbage collection inspectors in vehicles with cameras
Common citizens phone mobile cameras
Send images
Send images info
Lisbon collection teams
Trash classification algorithm
Images with trash identified

Figure 5. Garbage Identification System

The proposed system involves the submission of the images captured by the collection inspectors and those shared by citizens on the app 'My street' to the classification algorithm. The classification of each image is not done as a whole, but by blocks. Therefore, each block must be classified as trash or not. All blocks classified as trash are identified in the image as a result and in the end this information is sent to the collection teams.

This method would allow a real time quality control of waste collection in areas where waste production is excessive and a better management, in the city of Lisbon. In addition it will be possible to understand where and when to act in comparison to the current collection control procedure.

The first phase of the system implementation is data collection followed by pre-processing. Then this data goes to the waste detection and classification phase, using an algorithm already trained for this purpose.

It is expected that, for a block that does not contain trash, the algorithm associates the class 'no_trash' and for the block identified with trash, the class 'trash'. Figure 6 illustrates this process.

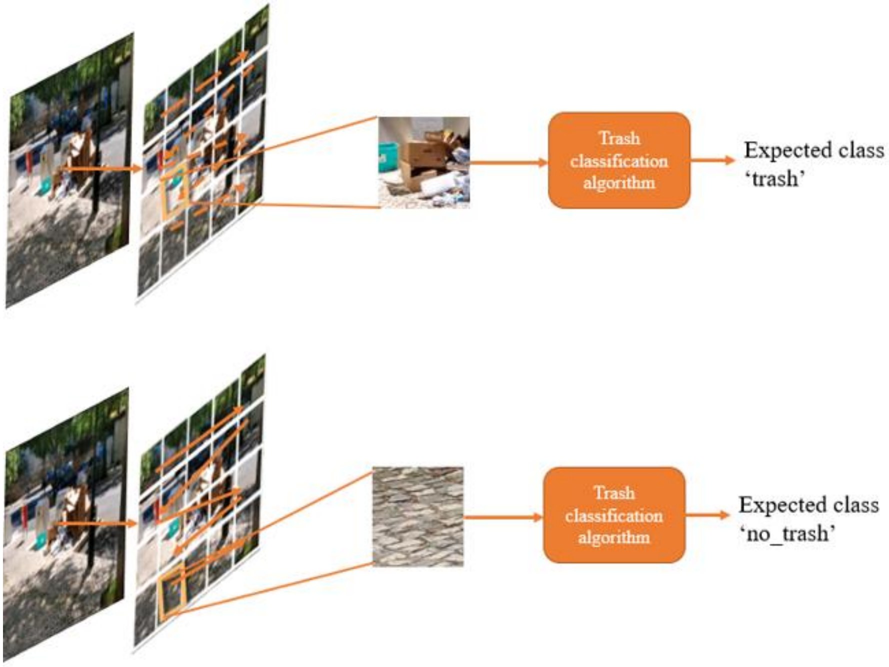

Trash classification algorithm
Expected class 'trash'
Trash classification algorithm
Expected class 'no_trash'

Figure 6. Algorithm classification

Finally, the output of the algorithm is an image with the identified residues, as depicted in Figure 7.

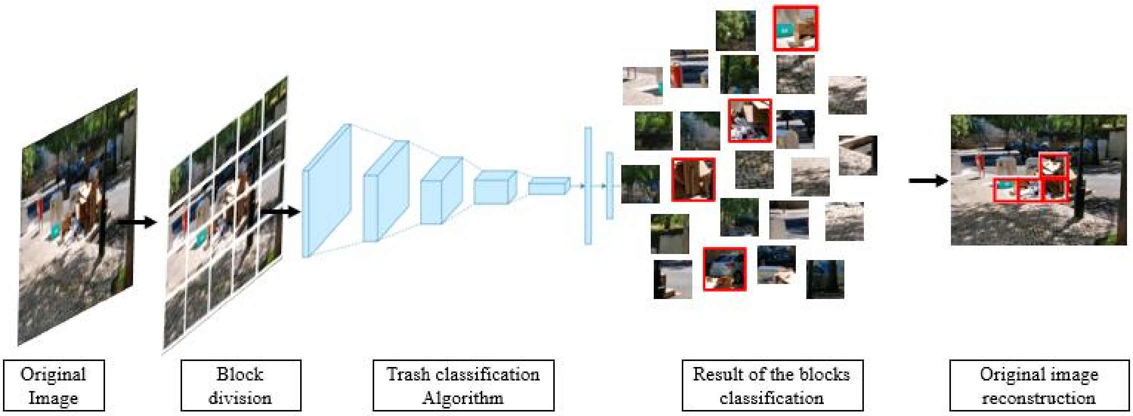

Original Image
Block division
Trash classification Algorithm
Result of the blocks classification
Original image reconstruction

Figure 7. Expected system behavior

The realization of this work involves collaboration between ISCTE-IUL and the Lisbon City Hall. This project is focused on the development of a binary classification algorithm for the recognition of residues in images. The shared images do not follow any acquisition rules. They were collected without any kind of control, increasing the complexity for achieving the classification algorithm goals.

The following section describes how the classification algorithm works.

3.2. Trash Classification Algorithm

Since during the research of works related to the subject under study no deep learning algorithms were found that dealt directly with the recognition of residues outside the equipment and most of the input images provided are large and obtained without any kind of control, it was decided to start from a simple architecture as a solution between the complexity of implementation and the time required for training and expected results. However, a solution to solve the problem mentioned above would involve research in other domains. As is the case of the project work [41] developed by the student Carolina Gonçalves, in which the main objective of the developed algorithm was to identify invasive species 'Acacia Longifolia'.

The classification algorithm received as input large resolution images previously divided into smaller sub images obtained from a drone. Based on a CNN architecture, the main function of this algorithm is to classify the images into two possible classes 'with' and 'without' invasive species. Since the architecture used in this project work is quite simple and achieved a high value with respect to the classification hit rate, it was decided to follow the same procedure for this project work. It was later modified and adapted according to the results obtained from the experiments performed for the specific case of this project work.

The initial CNN architecture used in this project is represented in Figure 8:

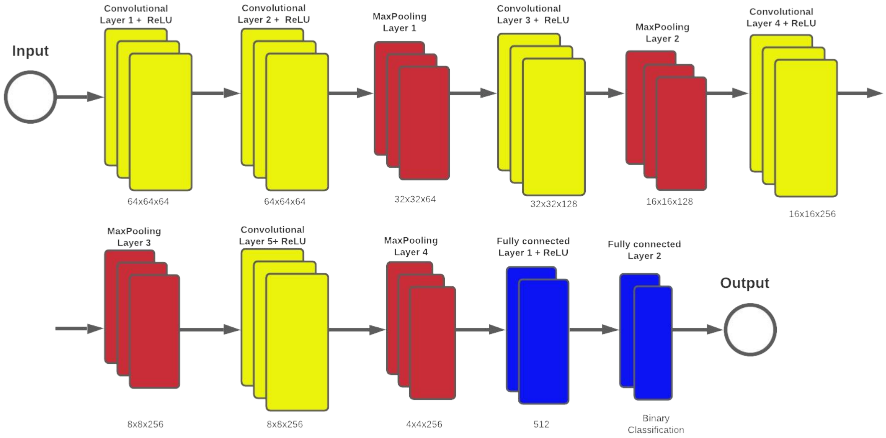


Figure 8. First arcquitecture setup

This was the first architecture developed. Its inputs are 64 by 64 pixel images. The kernel size for all convolutional layers is 3 except for the second convolutional layer which is 5. The MaxPooling layers with a kernel size of 2. Quite simple architecture containing about 5 convolutional layers interspersed by a Maxpooling layer configured with a stride of 2. The last two layers of the architecture, consisting of fully connected layers, which are basically the 'classical' classification layers based on neural networks. The last layer constitutes the output that allows classifying the ‘no_trash’ or ‘trash’ image. The results obtained in this configuration are explained in detail in the section 4.

3.3. Data Acquisition for system training

In this section the creation of the dataset is described, from obtaining the data to how it is treated and placed at the input for the algorithm training.

As mentioned before, the dataset used was provided through a repository of the Lisbon City Hall. Figure 9 demonstrates the flow from data source to algorithm training.

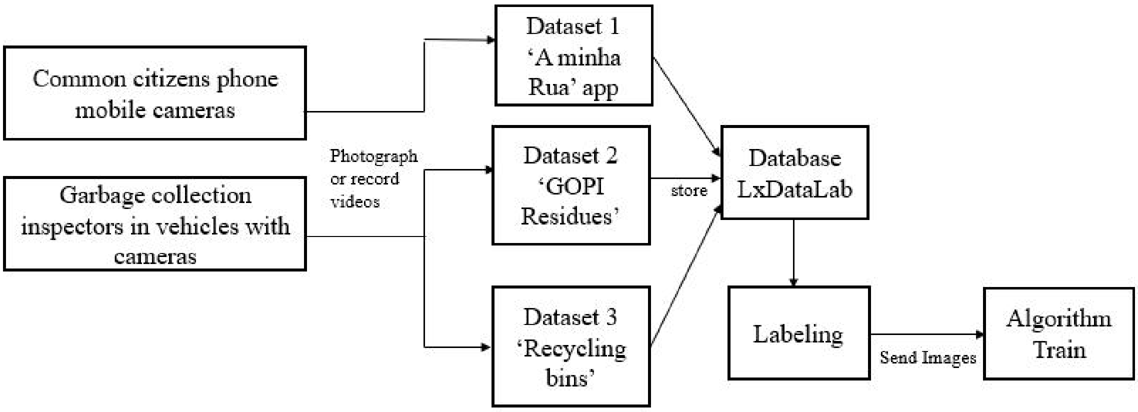

Common citizens phone mobile cameras
Garbage collection inspectors in vehicles with cameras
Photograph or record videos
Dataset 1 'A minha Rua' app
Dataset 2 'GOPI Residues'
Dataset 3 'Recycling bins'
store
Database LxDataLab
Labeling
Send Images
Algorithm Train

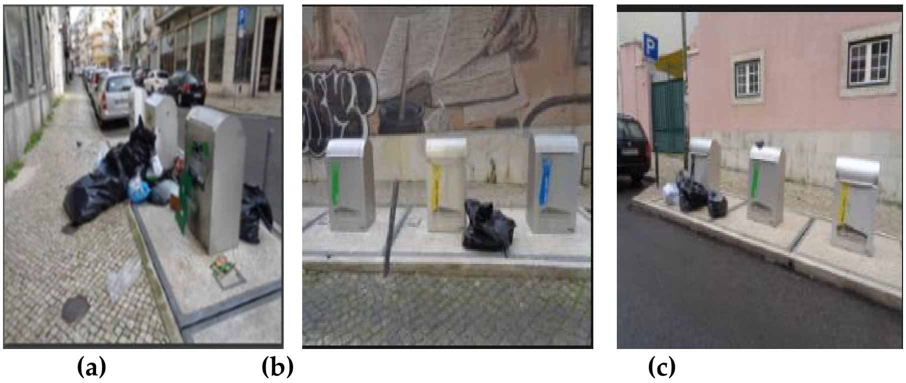
Figure 9. Data acquisition

The process begins with the collection of images by waste collection inspectors or ordinary citizens, of images of waste outside the equipment. These images, previously separated by dataset, constitute the 'LxDataLab' database. Access to the database allows this data to be tagged. Finally the training and testing for the different classification models is performed.

The following two sections explain what was done to generate the data that was used for training the CNN models proposed in the scope of this project.

3.3.1. Dataset

As explained before, the input data – images – is an essential requirement for the development of the garbage detection system based on supervised learning. Data is provided by the Lisbon City Hall, through a private repository called 'LxDataLab', managed by the Lisbon Center for Urban Management and Intelligence. This repository contains data regarding different problems where the use of machine learning may potentially be useful for task automation. For the misplaced garbage detection task, a set of images was collected from several sources. One of such sources is the app 'A minha Rua', where images were acquired by anonymous citizens. Most of these images consist of examples where the waste is deposited outside recycling garbage containers, such as those depicted in Figure 10. In these images, the type of waste that prevails are the common garbage bags, in Figure 10-b), d), e) and f), and cardboard/boxes, in Figure 10-a) and c.



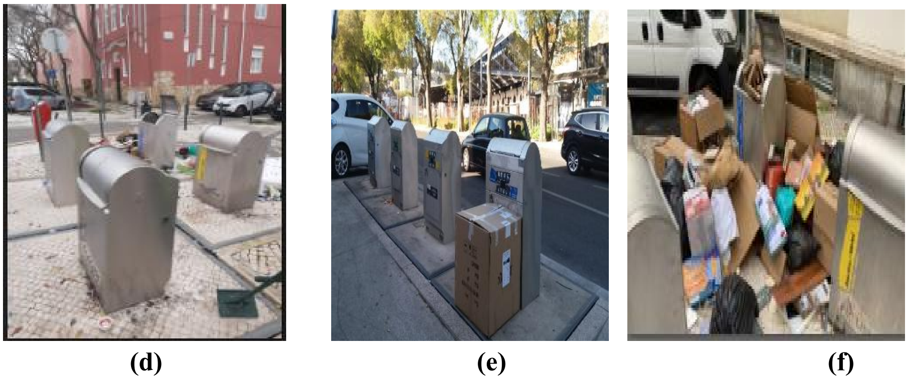


Figure 10. Pictures of waste residues from app 'A minha Rua'

Other image sources are smartphone cameras used by the garbage collection inspectors, who snapped images and filmed videos at key locations of Lisbon where trash disposal outside of equipment is prevalent, as shown in Figure 11. There is more variety in the type of waste and disposal equipment in these images, with large residues such as bulky trash in Figure 11 (labels a and f), biodegradable waste, as in Figure 11 (labels b and e), incorrectly deposited waste (label d), and finally places where the equipment cannot support the amount of waste produced (label c) and thus deposited outside the equipment.

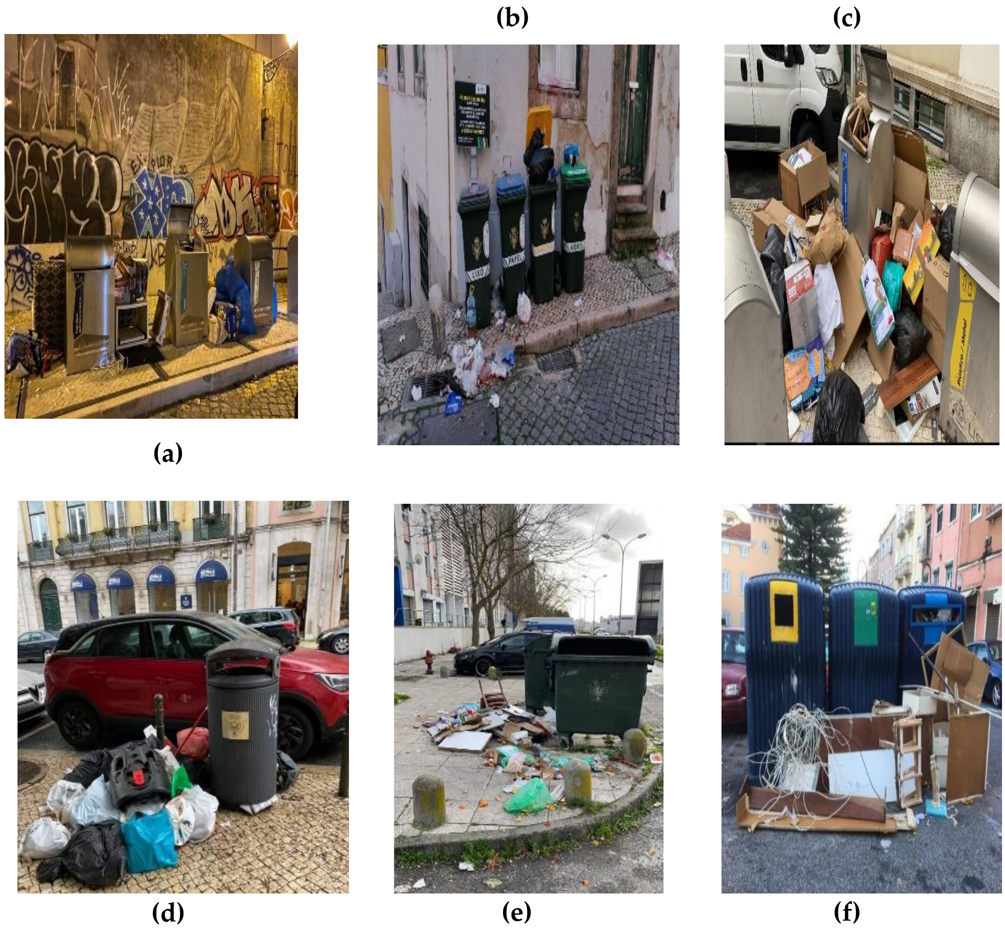
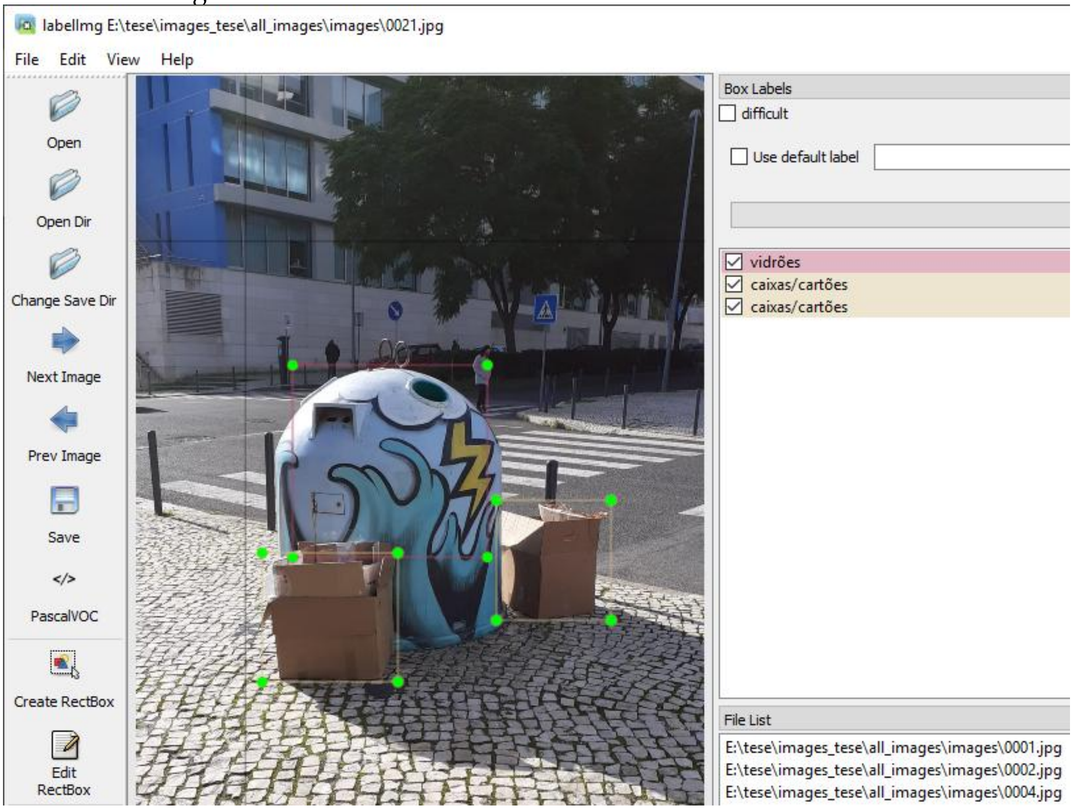


Figure 11. Images taken by the collection inspectors

With all these images collected, an unlabeled dataset was built by the LxDataLab team and shared in the scope of this thesis. As previously mentioned, after analyzing the shared images, it was concluded that they were obtained without any control. This led to extensive previous data preparation work.

The first step for the treatment of the images was to count and characterize each one of them. A total of about 1451 images were available. The waste collection inspectors provided 1032 images obtained via smartphone cameras from still positions; 259 were extracted from 5 videos acquired from moving vehicles in the city of Lisbon; and 160 images came from the app 'A minha Rua'.

The second step was the annotation of these images, by characterizing image elements such as: numbering that represents the id of the image, the typology of the containers equipment’s, the quantity which represents the residues amount, the type of waste, the location of the residues in the image, resolution and flagged garbage. Some of these characteristics could potentially contribute to a better classification model. The images were initially sorted out according to a new numbering system because their original numbering system was not normalized. The annotations were performed using the labelImg software, a graphic tool that allows an easier image annotation process. With the help of bounding boxes elements were identified as: boxes/cards, loose trash and bags, as shown in Figure 12.



labelImg E:\tese\images_tese\all_images\images\0021.jpg

□ difficult

□ Use default label

☑ vidrões

☑ caixas/cartões

☑ caixas/cartões

E:\tese\images_tese\all_images\images\0001.jpg
E:\tese\images_tese\all_images\images\0002.jpg
E:\tese\images_tese\all_images\images\0004.jpg

Figure 12. Labelling of image 0021.jpg with LabelImg

While performing the labeling process, it was found that there was a noticeable variance in the image resolution, illumination conditions and points of view, which could impose obstacles to the network's learning. The images included in the dataset also contain large portions of background that include elements such as sidewalks, buildings, vegetation, roads, and signs.

Given that a wider content variation on the image dataset could potentially lead to a better network generalization for detecting waste placed outside the equipment, it became critical to collect as many samples as possible. However, the amount of images depicting misplaced garbage was much larger than those without it. In order to overcome this issue, the classification it was decided to classify smaller image patches instead of providing a global classification for each image. The original photos were therefore split into smaller 64x64 sub-images, resulting in a greater number of samples that helped in overcoming the mentioned issues. For each sub-image, the number of the corresponding source image and the coordinates of the top-left sub-image pixel were stored in the filenames for future reference. Figure 13 shows examples of sub-images generated from the same source image.

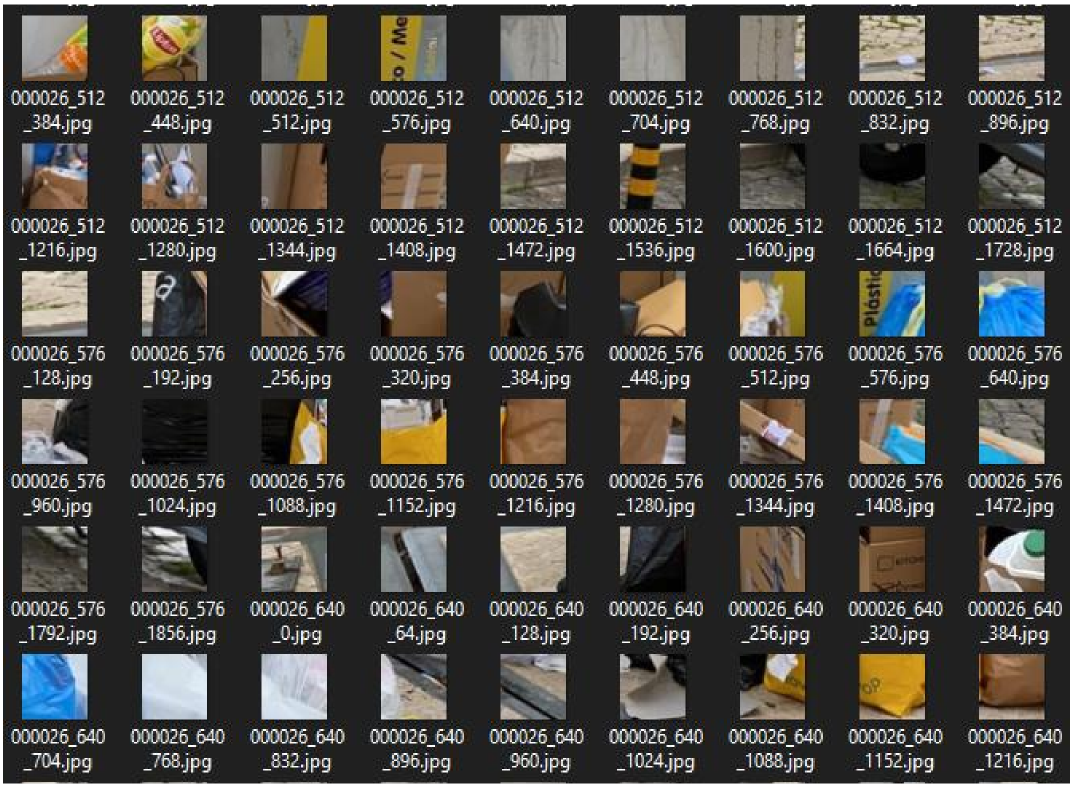


Figure 13. Creation of sub images from the original images

Each sub-image was then labeled as depicting thrash or no thrash ('trash' and 'no_trash' categories), and the sub-image dataset was organized according to those two categories.

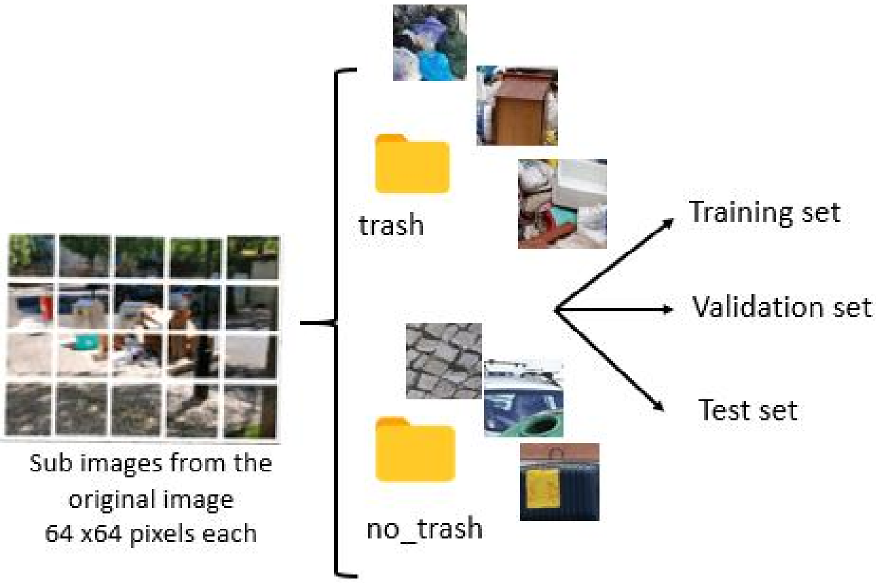


Figure 14. Class Distribution and Division of the dataset

After setting up the dataset, the sub-images were split into three sets of input data to prepare it for training, validation and testing of the CNN-based machine learning algorithms, as illustrated in Figure 14. The Training data set is made up of data that the model will use to train itself by matching the input to the expected output, which usually is the set with the largest amount of data samples. The validation data set is used for determining how well the training process is performing and can be used for detecting undesirable situations such as overfitting. Finally, the test dataset can be used for evaluating the performance of the model's predictions on new data that was not used during training.

Approximately 19738 samples were used in the final experimental phase, using a split 50/30/20 percent for training, validation, and test, respectively.

There were around 10427 samples in the training set, with 5214 in the 'trash' category and 5213 in the 'no_trash' category, representing 50% of the total. About 5167 samples were identified in the validation set, with 2585 categorized as 'trash' and 2584 as 'no_trash' representing 30% of the input data, and finally about 4144 samples were identified in the test set, with 2136 samples in the 'trash' category and 2008 in the 'no_trash' category, representing 20% of the input data.

3.3.2. Data Organization

The pre-processing technique consists in creating for each of the data sets, an array of features and labels that associates them. To do this, a method, 'getData(data_dir)' was developed. With this input parameter the expected result of this method is to return an array of arrays in which each array is composed by converting each RGB image into an array with the correct size, the features and the class/label it belongs to.

After applying this method to the 3 data sets previously mentioned, we have in the end generated 3 arrays that symbolize the data sets under study.

In order to organize the data structures that allow you to send data to the network for each of the data sets, is created an array of sub-images and their corresponding labels. To perform this task, the method, 'getData(data_dir)' was developed. The expected result of this method is to load into an array of arrays the various sub-images that are in the folder "data_dir" and corresponding labels.

This method is applied to the three datasets previously mentioned (training, validation and test), organized into the arrays as described.

3.3.3. Normalization and Data Augmentation

After generating the arrays, the input data is normalized. Normalization consists in defining the range of values in which the data will be transformed to. This process can avoid the saturated values in the activation functions [36]. Thus, the input image data was normalized to the range [0,1] dividing the RGB pixel values by 255.

With the arrays normalized to operate at the correct intervals, data augmentation was used to facilitate network training and to prevent overfitting. As explained at the beginning of the section, the variety and amount of subimages containing residues were smaller when compared with subimages containing no_trash. Therefore, it became necessary to apply techniques that would produce this variety. This was achieved by using data augmentation.

The problem of imbalanced data classification has been discussed by Kingma and J. Ba in [39]. It prompted us to increase the number of images in the class whose dataset scale was smaller than that of others. For augmenting image data, the generic practice is to perform geometric transformations, such as rotation, reflection, shift, and flip. However, images generated by a single type of operation are similar to each other. They increase the probability of overfitting. To avoid this situation, a new data augmentation method was proposed, which could randomly select operations and combine them together to produce new images. Available operations and values of them are shown in Table 1.

Table 1. Parameter set used for data augmentation cited.

Operation	Value
featurewise_center	False
samplewise_center	False
featurewise_std_normalization	False
samplewise_std_normalization	False

zca_whitening	False
rotation_range	30
zoom_range	0.2
width_shift_range	0.1
height_shift_range	0.1
horizontal_flip	True
Vertical_flip	False

4. CNN Model Training Experiments

To design and build the training and testing models, the Python programming language and the tensorflow/keras library were used. This library provides source code and allows for the rapid creation of code to train ML models. The code developed in the scope of this project was therefore written in Python, running on top of tensorflow, all on google Colab, a cloud service hosted by google that allows ML and AI research. The tensorflow/keras API version used was 2.5.0 and the Python version used was 3.7.11. Since the memory dedicated by google Colab was temporary, the memory dedicated for this project was all allocated to the pc's CPU, so the network training time was quite extensive.

For automatic recognition of residues in the images, two types of convolutional networks were developed and trained: a model built from scratch and a model in which the architecture is already preconfigured and available on keras, the MobileNetV2, thus avoiding the creation from scratch and training only the last layer for the intended purpose.

Another relevant variable in the performance of the model is the time it required for training, considering the available resources.

All the experiments were carried out using a HP Elitebook 820 G3, with 32 GB of RAM and an Intel ® CoreTM i7-6500U CPU @ 2.59GHz.

4.1. Hyperparameter Settings

The performance of neural networks with respect to classification is influenced by the values of their hyperparameters. Thus, several experiments were performed, always taking into account the variation of these network hyperparameters, until the final architecture was reached, with a desired hit rate and training execution time. The number of layers, the size of the convolution filters, the number of epochs per training, the probability of random deactivation of neurons in the network, the optimization algorithm and the value of the learning rate were varied.

The Table 2 and Table 3 show the configurations of the tests performed corresponding to the classification in two classes. The configuration and test architecture with the best hit rate was used to build the network that distinguishes, 'trash' and 'no_trash' classes. Table 2 describes the configurations of the CNN networks developed from scratch, then Table 3 describes the configurations performed on the CNN networks where transfer learning was used.

Table 2. Custom CNN Model Settings.

Experiment	Convolutional Layers	Filters Dimension	Max Pool Layer	Dropout	Ephocs	Optimization Algoritm	Dataset Images

		1st Layer	Other layers					Train	Validation
1	4	64	128 256 256	4	N/A	200	Adam	2855	2015
2	3	32	32 64	3	0.4	100		10469	5173
3	3	32	32 64	N/A				10427	5167
4	3	32	32 64	N/A				12510	7152
5	5	32	32 32 64 64	3				12510	7152

Experiment	Learning Rate	Activation Function	Training time	accuracy		Loss		Overfitting
				Train	Validation	Train	Validation	
1	10-3	sigmoid	6h	98%	96%	0.02	0.22	No
2	10-6	softmax	4h	76%	68%	0.53	0.60	No
3			3.5h	74%	66%	0.55	0.63	No
4			4.5h	76%	68%	0.51	0.60	Yes
5			2.5h	94%	87%	0.17	0.44	Yes

Table 3. Transfer Learning Model Settings.

Experiment	Model	Weights	Dense Layer	Dropout	Ephocs	Optimization Algoritm	Nº Dataset Images	
							Train	Validation
6	MobileNetV2	Imagenet	128 2	0.2	100	Adam	10469	5173
7			1	0.2	75		10427	5167
8			256 2	0.2	100		12510	7152
9			2	0.2	100		10427	5167

Experiment	Learning Rate	Activation Function	Training time	accuracy		Loss		Overfitting
				Train (TL)	Validation(TL)	Train (TL)	Validation(TL)	
2 –6	10exp-5	softmax	1h	89%	80%	0.28	0.42	No
3 -- 9		sigmoid	1.5h	99%	95%	0.04	0.22	Yes
4 -- 7		softmax	1.h	99%	95%	0.05	0.22	Yes
5 -- 8		softmax	1.5h	98%	88%	0.12	0.	Yes

4.2. Baseline Model

4.2.1. Architecture Description

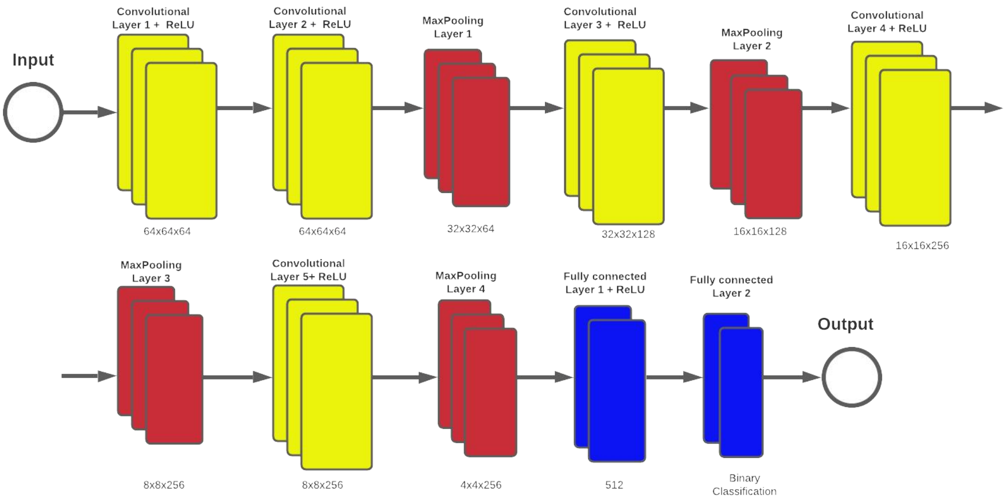


Figure 15. First Architecture setup

Figure 15 represents the first architecture developed. Quite simple architecture as explained before in section 3.2. It was necessary to define the optimization and loss functions. The optimization function chosen was Adam [39], because its main function is to measure the mean squared error between each input element, in this case the images, and the categories. This has an adjustable parameter, learning rate, and is mostly used to iteratively update the weights on the training data, in this first configuration the learning rate was set to 10-3. While the loss function used was the binary cross-entropy loss function, this allows to evaluate how good or bad the predicted probabilities are [40]. It should return low values when the neural network is performing good. The activation function used between the convolution and MaxPooling layers was the ReLU. In the last Fully Connected layer, the 'sigmoid' activation function is used since it is the output layer.

At the time this first experiment was done, the dataset did not have as many samples as in its final version. It consisted only of 4870 images. 2855 for the training set and 2015 for the validation set. It was decided to use this CNN architecture repeatedly with variations in the learning rate and the number of epochs with the goal of figuring out which was the best value. Initially the number of epochs configured was 200, however the best model result, i.e. with the lowest validation loss value, may occur before the 200th epoch.

The network is set up for training. In total its training was achieved in about 6h.

4.2.2. Results Evaluation

The results obtained can be observed in Figure 18, where the accuracy and loss metrics were measured. The main purpose of the accuracy metric is to calculate how often the predictions match the labels, while the loss metric is the result of a function that calculates cross-entropy loss between the ground-truth labels and predictions. In the figures shown, the x-axis represents the epochs while the y-axis represents accuracy and loss respectively.

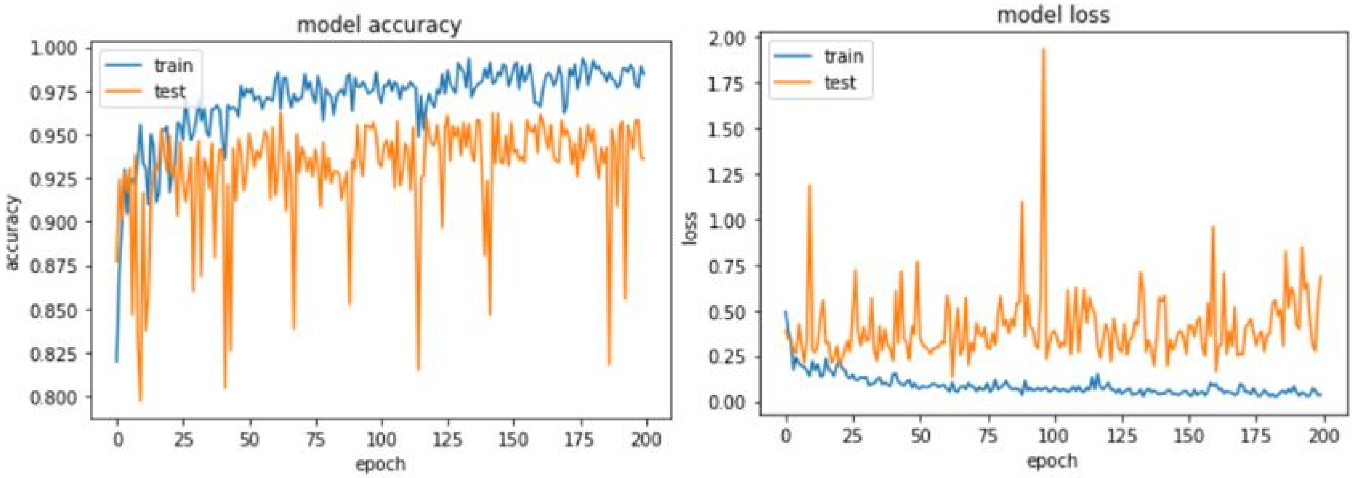


Figure 16. Model Training results – Experiment 1

Although an initial learning rate of 10-3 was used, a learning rate between 10-7 and 10 -3 was also tested in the same architecture, however not showing any difference in the final results in the accuracy and loss charts.

By analyzing the plots in Figure 16, it is possible to conclude that the training values for accuracy (represented in blue) reveal to be different from the validation values (represented in orange), the latter having shown several oscillations throughout the epochs, with values varying between about 80% and 96%. The same happens with the loss metric at the validation value level, with oscillations in the loss values, values varying between 0.22 and 2. Although the net was configured to save the best training results, through the graphs it is possible to conclude that the dataset was unrepresentative for each of the classes under study and too small.

Additional tests regarding the architecture of the CNN were also carried out with the goal of understanding the impact of the MaxPooling and Convolutional layers on the performance of the CNN, corresponding to experiments 2 and 3 in Table 3. In both experiments the dataset was increased to 10427 and 10469 samples respectively. A convolutional layer was removed and the filters were decreased. The first convolutional layer was left with 32 filters and the other two with 32 and 64 respectively. In the second experiment one Max Pooling layer was reduced in relation to the first architecture. In the third experiment the MaxPooling layer was removed. A dropout of 40% was added and activation fuction were changed to softmax. Note that the data augmentation technique was initially done both on the training and validation set. For these approaches it was decided to do only on the training set. Also, the number of epochs was decreased to 100 for these experiments. The results were very similar. Huge difference in result graphics in comparison with first Architecture model. However the dataset seems to be not representative as much as pretended, when tested with real images that were not used during model training. To get a model that better classified residues, more images were added to the dataset, and the fourth experiment was performed.

After the set of experiments performed it was concluded that the model that would be more suitable for recognition of garbage deposited outside the garbage equipment was the model configured in experiment 4 followed by the model with transfer learning corresponding to experiment 8 in Table 3.

The next sections explain the last architecture modifications performed where the desired results were achieved.

4.3. *Final CNN Model*

4.3.1. Architecture Description

Figure 17 represents the last architecture developed and trained from scratch.

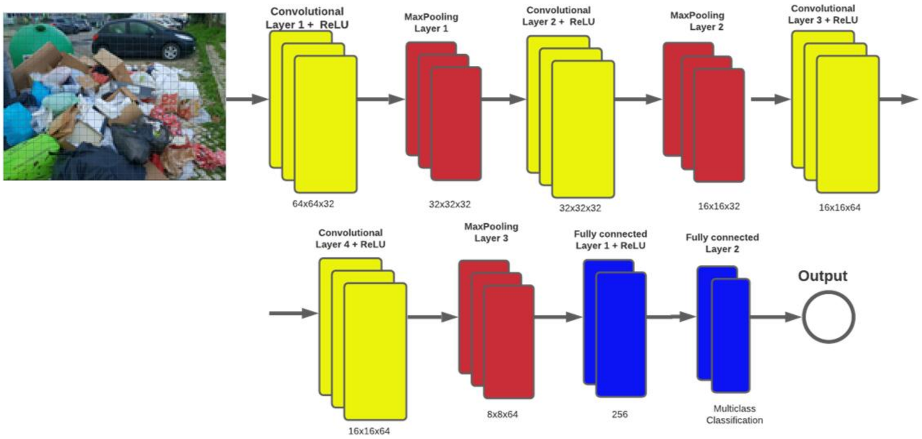

Convolutional Layer 1 + ReLU
MaxPooling Layer 1
Convolutional Layer 2 + ReLU
MaxPooling Layer 2
Convolutional Layer 3 + ReLU
64x64x32
32x32x32
32x32x32
16x16x32
16x16x64
Convolutional Layer 4 + ReLU
MaxPooling Layer 3
Fully connected Layer 1 + ReLU
Fully connected Layer 2
Output
16x16x64
8x8x64
256
Multiclass Classification

Figure 17. Last model architecture

In the training options of the CNN, a Learning Rate of 0.000001 is set, at the end the code, as we can see in Figure 17, and returns a model with 1,124,162 trainable params. The batch size is set to 32, in this way there are 391 iterations per epoch, where each batch corresponding to 3% of the total size of the training database. Based on training tests, 100 epochs are established, being insufficient to obtain adequate learning in the network.

4.3.2. Results Evaluation

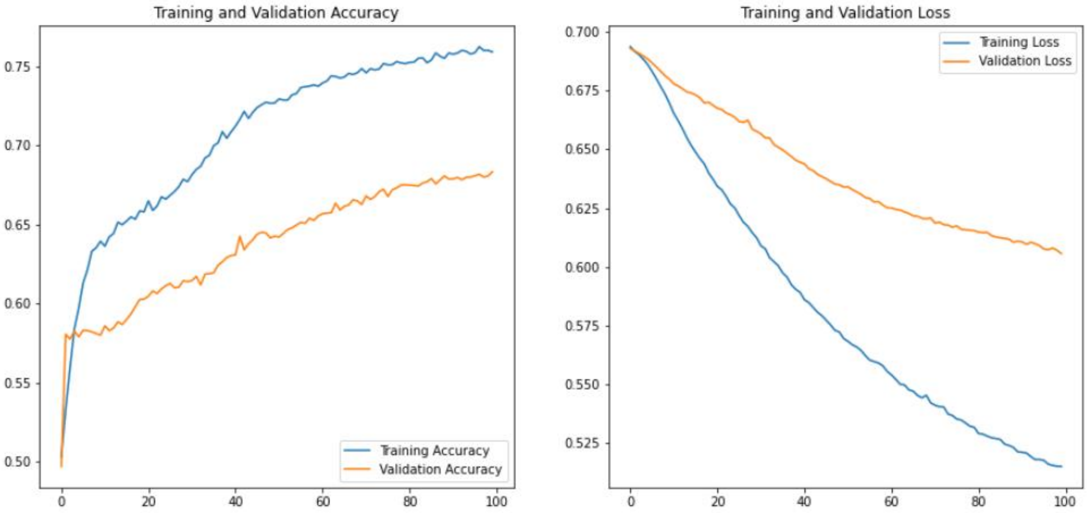


Figure 18. Model Training results – Experiment 4

In Figure 18, the training of CNN results is shown. The left graph represents the accuracy in the classification of the training images in each of the periods. In blue color the checks of the training images are presented, and the orange color are of the validations. 76% accuracy was obtained in the last epoch with the training images and 68% with the validation images. The right graph shows the losses by epoch, in blue the losses are with the training images and the orange correspond to the validations, where losses of 0.51 and 0.60 are gotten in the last period, respectively.

In order to improve the results obtained we opted to use transfer learning, Figure 9. For this we used the 'MobileNetV2' architecture available in Tensorflow's Keras. This model is implemented to accept images in which the maximum allowed pixels are 224 x 224, which led us to adapt it to accept the size of the images under study (64 x 64 pixels) through the variable 'input_tensor'. The weights that this model uses when imported into Tensorflow are those from 'imagenet'. Transfer learning is used in this case to speed up the training of the network.

4.4.Transfer Learning

4.4.1. Architecture Description

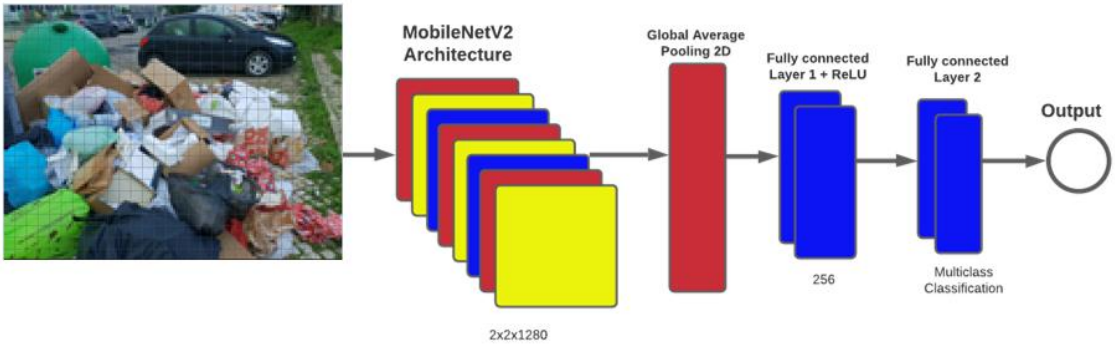


Figure 19. MobileNetV2 architecture

In order to improve the results obtained and have a comparison model, a transfer learning approach was opted, Figure 19. For this, the 'MobileNetV2' architecture available in Tensorflow's Keras was used. This model is implemented to accept images in which the maximum allowed pixels are 224 by 224, which led us to adapt it to accept the size of the images under study (64 by 64 pixels) through the variable 'input_tensor'. The weights that this model uses when imported into tensorflow are those from 'imagenet'. Imagenet is a database of images that contains classifications for these images. Transfer learning is used in this case to speed up the training of the network. Having imported the base model, it then remains to define which layers should be trained, adapting the model to the classification problem under study. A GlobalAveragePooling2D layer is added whose main function is to convert the features into a single vector per image. A 20% dropout was added to the model. The activation function used between the convolution layers and the first fully connected layer is ReLU with 256 neurons. A dense layer with 2 neurons and whose activation function is 'softmax'. We followed the same logic as the previous experiment, at the hyperparameter level: Adam optimization function with learning_rate =10-5; loss function set to 'Sparse CategoricalCrossentropy. The number of epochs as in the previous experiments of 100. The dataset used was the same for each of the sets.

4.4.2. Results Evaluation

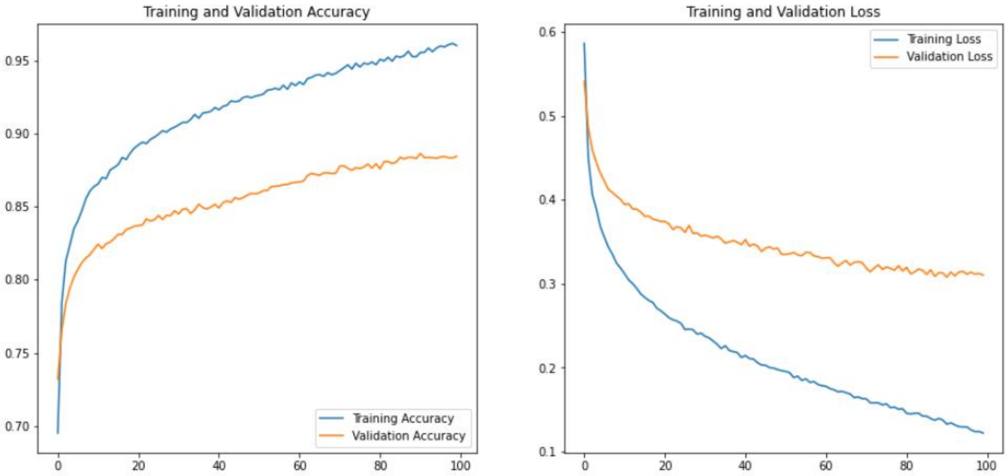


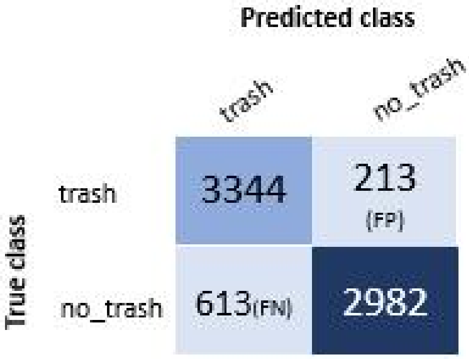
Figure 20. Accuracy and Loss Results

Is possible to conclude that, analyzing Figure 20, in comparison with results obtained in previous experiments, despite some oscillation in the curves, both in terms of accuracy and loss, better results were obtained. For the training set there was an exponential increase in accuracy, reaching 98%, while for the validation set there was a slight increase, but from epoch 80 it remains constant, but reaching 80%. While in the loss metric, for the training set there was a sharp and exponential decrease as expected, reaching 0.12, a very good value, and the validation set varied in 0.30, but with constant values between the 60th and 90th epochs. The Table 4 shows the performance of the last architecture models trained.

Table 4. Training performance of selected models.

Model Name	Training Accuracy (%)	Validation Accuracy (%)	Training Time (ms)/Batch Size
CNN_Model_trash	76	68	332
MobileNet-v2	98	80	160

By generating the classification report and the confusion matrix it was also possible to draw some conclusions. The Figure 21 describes the confusion matrix for the validation set.



		Predicted class	
		trash	no_trash
True class	trash	3344	213 (FP)
	no_trash	613(FN)	2982

Figure 21. Matrix Confusion for validation set results – Experiment 8

It is possible to conclude after analyzing the Figure 21, that with the increase of variety and number of images, that classification done by the model is better than the previous experiment, although there are some false positives that was what was intended to decrease with this experiment. Only 6% were incorrectly classified by the model of the images that existed in the class 'trash'. As for the set of images that represent the 'no_trash' class, 2982 images out of 3595 belonging to the set were correctly classified by the model. This allows us to conclude that it is still not the perfect model, even with the increase in accuracy and decrease in loss, there will be parts of the original image that will be misclassified.

The Table 5 describes the values obtained in report classification for the metrics precision, recall and F1-score for the test set. The values were obtained considering that the 'trash' class is the positive class and the 'no_trash' class is the negative class.

Table 5. Precision, Recall and f1-score for previous test – Experiment 8

Precision	Recall	F1-Score
0.94	0.85	0.89

From the analysis of the Table 5, we can mention that for this experiment, the perfect value was not achieved in any of the metrics, however the recall reaching 85% for the 'trash' class and 94% or precision which allow to conclude that the system can recognize trash in image but some are misclassified.

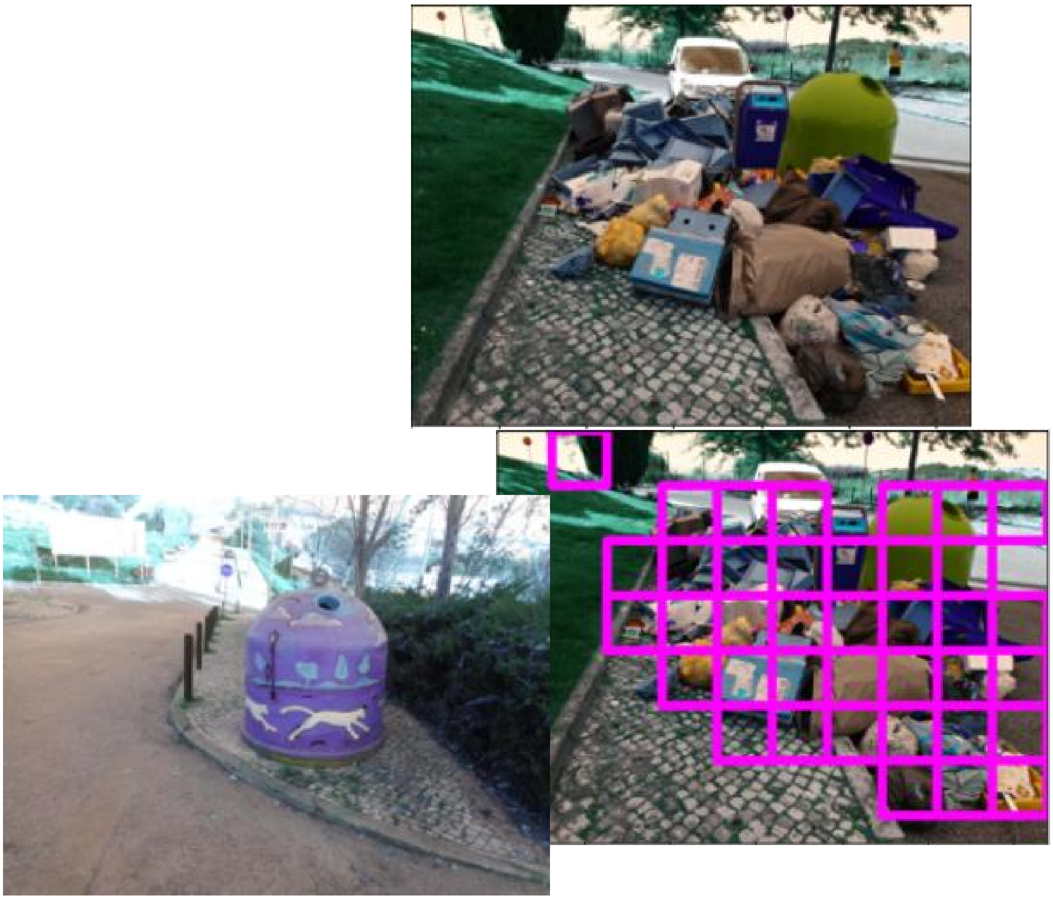
5. Full Image Classification Tests

After studying the trained model, where it was found that the values were meeting the expected, it was decided to test the model on real images. As explained initially, after training the model, it is deployed on the proof of concept under study and tested on real scenarios. At the code level the following sequence of steps was followed:

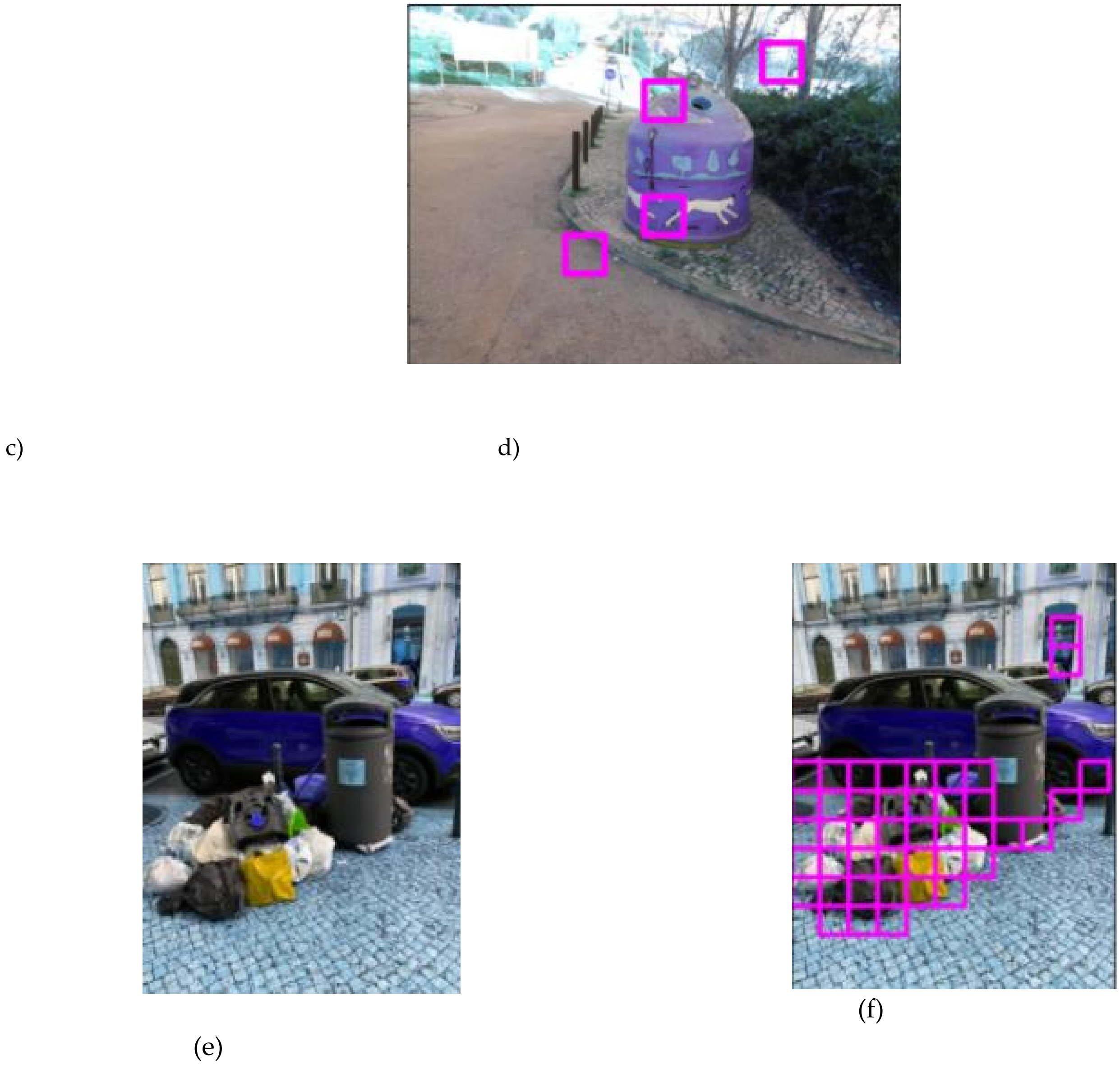
First the original images are placed in a folder (images collected by the camera inspectors without any treatment), then these images are subdivided into images to be received by the already trained model, i.e. into sub-images of 64 by 64 pixels. The third step is to submit these sub-images to the model for classification. Once the entire set of sub -images is classified to the corresponding original image, the system returns this original image again complete and with garbage properly marked. Represented by squares of 64 by 64 pixels. As mentioned earlier, the model goes through 64 by 64 pixels of the original image and classify each of them. Whatever is classified as 'trash' is identified in the original image with a colored square.

The following images, in Figure 22 represent three images classified by the prototype under study: the left side depicts the original images; the right side depicts the images after the classification.

a)



b)



c) d)

(f)

(e)

Figure 22. Garbage System residues Identification results – Experiment 8.

As observed in the images on the right side it is possible to verify that, in relation to the previous experiment, although the results were better, the classification is more concentrated in the residues. There are still some sub images classified as 'trash', which was also expected since when obtaining the confusion matrix there was a percentage of false positives, although small, such in the cases of Figure 22-b) and d). It should be noted that even with the increase of images for a better representation of the universe of images corresponding to the classes under study, there were still cases in which windows, tires were identified as 'trash', for instance, Figure 22-f).

Based on the results obtained, both from accuracy and confusion matrix as well as the classification report, it was concluded that the architecture presented in 4.4 section would be the best architecture for recognizing waste outside the designated equipment.

Note that the results obtained in the experiments presented in the configuration tables decreased as the experiments were performed. This is because although the results were getting worse in terms of metrics, the tests on cases were more representative of the intended goal.

It is possible to conclude that the origin of the possible results could be in the dataset used. Considering the origin and the variety of image sizes. It is also important to point out that the results obtained in this project work represent few cases of the possible existing ones. It was found that texture and color factors predominated in the network at the time of classification. From the results obtained in these experiments, it is possible to conclude that many of the images that the camera submits in this prototype created is misclassified, because the system still does not consider a range of factors such as: classification by type of residue, size of the input images, illumination, texture among others. This adapted multiclass classification algorithm acquired knowledge through the data sets created and was optimized considering the problem that was intended to be solved, using Computer Vision techniques.

6. Conclusions

In this paper, a deep learning framework for the recognition of waste deposited outside the equipment using computer vision techniques has been proposed.

Two models were tested, the first, was a pre-trained model, MobileNetV2, and the second was built from scratch and fully trained.

The analysis of the results obtained in each of the experiments allowed us to assert if the trained model was meeting the desired requirements. During the first development cycles several changes were performed, both in the architecture and in the dataset. Section 4 also presents mechanisms for solving the problems encountered. When obtaining a model that met the intended requirements, we validated the model using real cases.

Regarding the first objective of this work: "being able to classify images from different sources", the results are encouraging, although images required extensive prior treatment before they were presented to the network, since they were obtained without any control, and they were of varying sizes. The process shown enables the creation of an effective training-set from which a reasonably accurate classification rule can be learned.

Estimating the amount of garbage deposited outside the garbage facilities was another proposed goal. It was successful because a relatively low false positive rate was achieved.

Improving the classification architecture and continuously updating the datasets became essential to achieve better results for identifying garbage in the images. The datasets had a preponderant role in this sense. Since the first dataset created led to worse results due to its lack of variety of examples of images with trash and images without trash. Limitations as loss of image quality after resizing the data. However, after several experiments, it led to having to build datasets from scratch, more realistic and somehow better represent each of the classes.

These changes in the datasets meant that we had to adjust parameters in the network. The results obtained show that it was indeed possible to adapt a multiclass classification algorithm based on deep learning to this specific problem.

Regarding the questions proposed at the beginning of this research, it is possible to conclude the following:

RQ1 – Is classification of residues in images better with a pre-trained CNN or through a network built and customized from scratch?

Yes. With the use of pre-trained networks for recognizing residues in the images, better results were obtained compared to the custom root architectures.

RQ2 – How close will the developed algorithm be to the accuracy rate of humans?

Taking into account the experiments performed it is possible to conclude that the configured algorithm in terms of hit rate, is not yet at the level of accuracy of humans. Even though the hit rate for both the validation set and the training set were higher than 80% and 98% respectively, when tested in real scenarios.

It is possible to conclude that the prototype can identify the residues in the images, after analyzing them, however there is still much work to be done to make the system autonomous. This research allows us to conclude that it is possible, with the help of computer vision techniques, to classify images from different sources and dimensions with the use of the correct architecture. It also allows us to conclude that segmenting the images into smaller images can solve the shortage of data issues. At this moment it is still far from the desired because, only shared photos and by themselves, do not solve the problem in depth. Therefore this work, comes to give the first contributions in that direction.

The experiments performed reflected in the results obtained have been positive, although there are still errors. The next step to improve these results should be to improve the data sets by collecting more data. Also a more accurate labelling differentiating different types of garbage, would be likely to improve results. A separation of the samples/images by class and resolution is also beneficial, as it is possible that images of various resolutions could compromise the results obtained. With these dataset upgrades and retraining the entire CNN we believe results could be improved. Segmenting the images at the network input may still be a good approach for both training and classifying the images, however experimenting with increasing the size of these input blocks may also be a viable alternative to improve results. In terms of improvement for the prototype, it should be able to do a more specific waste classification, i.e. classify by type of waste, such as: cardboard/boxes; bulky trash; etc. Another idea would be to implement a real-time strategy to estimate waste production in areas where dumping is excessive, based on the teams collection history. Create a waste management app in the areas where there is excessive deposition in the city of Lisbon. This app would be able to update the volume of waste automatically according to a previously configured time interval. We hope that this work has helped to take another small step towards cleaner and healthier cities.

Supplementary Materials: CML dataset.

Author Contributions: S.H.F. was responsible for system development, T.B and L.N guided and supervised the writing and management of the work, and J.C.F. coordinated the research and contributed to the article's writing. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Foundation for Science and Technology (FCT) through ISTAR-IUL's project UIDB/04466/2020 and UIDP/04466/2020.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All the data can be used on request.

Acknowledgments: This work was partly funded through national funds by FCT - Fundação para a Ciência e Tecnologia, I.P. under projects UIDB/EEA/50008/2020 (Instituto de Telecomunicações) and UIDB/04466/2020 (ISTAR)

Conflicts of Interest: The authors declare no conflict of interest.

References

1. M. S. Rad, A. von Kaenel, A. Droux, F. Tieche, N. Ouerhani, H. K. Ekenel, and J. P. Thiran, "A computer vision system to localize and classify wastes on the streets," Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 10528 LNCS, pp. 195–204, 2017, doi: 10.1007/978-3-319-68345-4_18.
2. B. J. Fonseca, D. M. A. Felermino, and S. M. Saide, "A deep convolutional neural network for classifying waste containers as full or not full," 2019, pp. 54–59. doi: 10.1109/ISC246665.2019.9071746.
3. C. Zhihong, Z. Hebin, W. Yanbo, L. Binyan, and L. Yu, "A vision-based robotic grasping system using deep learning for garbage sorting," 2017, pp. 11223–11226. doi: 10.23919/ChiCC.2017.8029147.
4. D. Fox, J. Sillito, and F. Maurer, "Agile methods and user-centered design: How these two methodologies are being successfully integrated in industry," 2008, pp. 63–72. doi: 10.1109/Agile.2008.78.

5. L. Donati, T. Fontanini, F. Tagliaferri, and A. Prati, “An Energy Saving Road Sweeper Using Deep Vision for Garbage Detection,” Applied Sciences, vol. 10, no. 22, Art. no. 22, Jan. 2020, doi: 10.3390/app10228146.
6. [6] P. Tiyajamorn, P. Lorprasertkul, R. Assabumrungrat, W. Poomarin, and R. Chancharoen, “Automatic Trash Classification using Convolutional Neural Network Machine learning,” in 2019 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM), Bangkok, Thailand, Nov. 2019, pp. 71–76. doi: 10.1109/CIS-RAM47153.2019.9095775.
7. Z. Wang, B. Peng, Y. Huang, and G. Sun, “Classification for plastic bottles recycling based on image recognition,” Waste Management, vol. 88, pp. 170–181, 2019, doi: 10.1016/j.wasman.2019.03.032.
8. R. A. Aral, S. R. Keskin, M. Kaya, and M. Haciömeroğlu, “Classification of TrashNet Dataset Based on Deep Learning Models,” 2019, pp. 2058–2062. doi: 10.1109/BigData.2018.8622212.
9. S. L. Rabano, M. K. Cabatuan, E. Sybingco, E. P. Dadios, and E. J. Calilung, “Common garbage classification using mobilenet,” presented at the 2018 IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management, HNICEM 2018, 2018. doi: 10.1109/HNICEM.2018.8666300.
10. “Commonly Used Machine learning Algorithms | Data Science,” Analytics Vidhya, Sep. 08, 2017. https://www.analyticsvidhya.com/blog/2017/09/common-machine-learning-algorithms/ (accessed Jul. 27, 2021).
11. S. Frost, B. Tor, R. Agrawal, and A. G. Forbes, “CompostNet: An Image Classifier for Meal Waste,” presented at the 2019 IEEE Global Humanitarian Technology Conference, GHTC 2019, 2019. doi: 10.1109/GHTC46095.2019.9033130.
12. D. O. Melinte, D. Dumitriu, M. Mărgăritescu, and P.-N. Ancuţa, “Deep Learning Computer Vision for Sorting and Size Determination of Municipal Waste,” Lecture Notes in Networks and Systems, vol. 85, pp. 142–152, 2020, doi: 10.1007/978-3-030-26991-3_14.
13. J. Liu and Y. Jiang, “Design of intelligent trash can be based on machine vision,” 2020, vol. 11584. doi: 10.1117/12.2579291.
14. Y. Liu, S. Zhong, Z. Tian, and K. He, “Design of Vision Servo Sorting Robot System Based on SVM,” J. Phys.: Conf. Ser., vol. 1550, p. 022032, May 2020, doi: 10.1088/1742-6596/1550/2/022032.
15. A. Hevner and Chatterjee, Design Research in Information Systems: Theory and Practice, vol. 22. 2010. doi: 10.1007/978-1-4419-5653-8.
16. M. Valente, H. Silva, J. Caldeira, V. Soares, and P. Gaspar, “Detection of Waste Containers Using Computer Vision,” Applied System Innovation, vol. 2, p. 11, Mar. 2019, doi: 10.3390/asi2010011.
17. H. Wang, “Garbage recognition and classification system based on convolutional neural network vgg16,” 2020, pp. 252–255. doi: 10.1109/AEMCSE50948.2020.00061.
18. G. Thung, garythung/trashnet. 2021. Accessed: Jul. 27, 2021. [Online]. Available: https://github.com/garythung/trashnet
19. M. O. Rahman, A. Hussain, E. Scavino, H. Basri, and M. A. Hannan, “Intelligent computer vision system for segregating recyclable waste papers,” Expert Systems with Applications, vol. 38, no. 8, pp. 10398–10407, 2011, doi: 10.1016/j.eswa.2011.02.112.
20. A. Salmador, J. Cid, and I. Novelle, “Intelligent Garbage Classifier,” International Journal of Interactive Multimedia and Artificial Intelligence, vol. 1, pp. 31–36, Dec. 2008.
21. A. Torres Garcia, O. Rodea-Aragón, O. Longoria-Gandara, F. Sánchez-García, and L. González-Jiménez, “Intelligent Waste Separator,” Computacion y Sistemas, vol. 19, pp. 487–500, Sep. 2015, doi: 10.13053/CyS-19-3-2254.
22. Y. Chu, C. Huang, X. Xie, B. Tan, S. Kamal, and X. Xiong, “Recycling,” Computational Intelligence and Neuroscience, vol. 2018, pp. 1–9, Nov. 2018, doi: 10.1155/2018/5060857.
23. L.-G. Omar, R.-A. Oscar, T.-G. Andres, and S.-G. Francisco, “Multimedia inorganic waste separator,” in 2013 IEEE International Conference on Multimedia and Expo Workshops (ICMEW), Jul. 2013, pp. 1–4. doi: 10.1109/ICMEW.2013.6618314.
24. W.-L. Mao, W.-C. Chen, C.-T. Wang, and Y.-H. Lin, “Recycling waste classification using optimized convolutional neural network,” Resources, Conservation and Recycling, vol. 164, 2021, doi: 10.1016/j.resconrec.2020.105132.
25. C. J. Baby, H. Singh, A. Srivastava, R. Dhawan, and P. Mahalakshmi, “Smart bin: An intelligent waste alert and prediction system using machine learning approach,” in 2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), Mar. 2017, pp. 771–774. doi: 10.1109/WiSPNET.2017.8299865.
26. S. N., P. M. Fathimal, R. R., and K. Prakash, “Smart Garbage Segregation amp; Management System Using Internet of Things(IoT) amp; Machine learning(ML),” in 2019 1st International Conference on Innovations in Information and Communication Technology (ICIICT), Apr. 2019, pp. 1–6. doi: 10.1109/ICIICT1.2019.8741443.
27. M. Arebey, M. A. Hannan, R. A. Begum, and H. Basri, “Solid waste bin level detection using gray level co-occurrence matrix feature extraction approach,” Journal of Environmental Management, vol. 104, pp. 9–18, Aug. 2012, doi: 10.1016/j.jenvman.2012.03.035.
28. G. Mittal, K. B. Yagnik, M. Garg, and N. C. Krishnan, “SpotGarbage: Smartphone app to detect garbage using deep learning,” 2016, pp. 940–945. doi: 10.1145/2971648.2971731.
29. J. E. T. Akinsola, “Supervised Machine learning Algorithms: Classification and Comparison,” International Journal of Computer Trends and Technology (IJCTT), vol. 48, pp. 128–138, Jun. 2017, doi: 10.14445/22312803/IJCTT-V48P126.
30. R. Briner and D. Denyer, “Systematic Review and Evidence Synthesis as a Practice and Scholarship Tool,” in Handbook of evidence-based management: Companies, classrooms and research, 2012, pp. 112–129. doi: 10.1093/oxfordhb/9780199763986.013.0007.

31. A. Liberati et al., "The PRISMA Statement for Reporting Systematic Reviews and Meta-Analyses of Studies That Evaluate Health Care Interventions: Explanation and Elaboration," PLOS Medicine, vol. 6, no. 7, p. e1000100, Jul. 2009, doi: 10.1371/journal.pmed.1000100.
32. I. Salimi, B. S. Bayu Dewantara, and I. K. Wibowo, "Visual-based trash detection and classification system for smart trash bin robot," in 2018 International Electronics Symposium on Knowledge Creation and Intelligent Computing (IES-KCIC), Oct. 2018, pp. 378–383. doi: 10.1109/KCIC.2018.8628499.
33. S. Shalev-Shwartz and S. Ben-David, Understanding Machine learning: From Theory to Algorithms. Cambridge: Cambridge University Press, 2014. doi: 10.1017/CBO9781107298019.
34. F. Heinrichs, "Using CRISP-DM to Grow as Data Scientist," Medium, Dec. 03, 2020. https://towardsdatascience.com/using-crisp-dm-to-grow-as-data-scientist-a07ce3fd9d56 (accessed Sep. 01, 2021).
35. M. Hassaballah and A. I. Awad, Deep Learning in Computer Vision: Principles and Applications. 2020. doi: 10.1201/9781351003827.
36. L. Bottou, "Large-Scale Machine learning with Stochastic Gradient Descent," undefined, 2010, Accessed: Sep. 01, 2021. [Online]. Available: https://www.semanticscholar.org/paper/Large-Scale-Machine-Learning-with-Stochastic-Bottou/fbc6562814e08e416e28a268ce7beeaa3d0708c8
37. K. Gurney, "Introduction to Neural Networks." Taylor & Francis, Oxford.
38. D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," ICLR, 2015.
39. U. Ruby and V. Yendapalli, "Binary cross entropy with deep learning technique for Image classification," International Journal of Advanced Trends in Computer Science and Engineering, vol. 9, Oct. 2020, doi: 10.30534/ijatcse/2020/175942020.
40. C. Gonçalves, "Identificação Automática de Plantas Invasoras em Imagens Aéreas"
Master thesis, Telecommunications and Computer Eng., ISCTE-IUL, Lisbon, 2019.
41. D. Powers, "Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation," Mach. Learn. Technol., vol. 2, Jan. 2008.

References

[1] M. S. Rad, A. von Kaenel, A. Droux, F. Tieche, N. Ouerhani, H. K. Ekenel, and J. P. Thiran, “A computer vision system to localize and classify wastes on the streets,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 10528 LNCS, pp. 195–204, 2017, doi: 10.1007/978-3-319-68345-4_18.

[2] B. J. Fonseca, D. M. A. Felermino, and S. M. Saide, “A deep convolutional neural network for classifying waste containers as full or not full,” 2019, pp. 54–59. doi: 10.1109/ISC246665.2019.9071746.

[3] C. Zhihong, Z. Hebin, W. Yanbo, L. Binyan, and L. Yu, “A vision-based robotic grasping system using deep learning for garbage sorting,” 2017, pp. 11223–11226. doi: 10.23919/ChiCC.2017.8029147.

[4] D. Fox, J. Sillito, and F. Maurer, “Agile methods and user-centered design: How these two methodologies are being successfully integrated in industry,” 2008, pp. 63–72. doi: 10.1109/Agile.2008.78.

[5] L. Donati, T. Fontanini, F. Tagliaferri, and A. Prati, “An Energy Saving Road Sweeper Using Deep Vision for Garbage Detection,” *Applied Sciences*, vol. 10, no. 22, Art. no. 22, Jan. 2020, doi: 10.3390/app10228146.

[6] P. Tiyajamorn, P. Lorprasertkul, R. Assabumrungrat, W. Poomarin, and R. Chancharoen, “Automatic Trash Classification using Convolutional Neural Network Machine learning,” in *2019* IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM), Bangkok, Thailand, Nov. 2019, pp. 71–76. doi: 10.1109/CIS-RAM47153.2019.9095775.

[7] Z. Wang, B. Peng, Y. Huang, and G. Sun, “Classification for plastic bottles recycling based on image recognition,” *Waste Management*, vol. 88, pp. 170–181, 2019, doi: 10.1016/j.wasman.2019.03.032.

[8] R. A. Aral, S. R. Keskin, M. Kaya, and M. Haciömeroğlu, “Classification of TrashNet Dataset Based on Deep Learning Models,” 2019, pp. 2058–2062. doi: 10.1109/BigData.2018.8622212.

[9] S. L. Rabano, M. K. Cabatuan, E. Sybingco, E. P. Dadios, and E. J. Calilung, “Common garbage classification using mobilenet,” presented at the 2018 IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management, HNICEM 2018, 2018. doi: 10.1109/HNICEM.2018.8666300.

[10] “Commonly Used Machine learning Algorithms | Data Science,” *Analytics Vidhya*, Sep. 08, 2017. https://www.analyticsvidhya.com/blog/2017/09/common-machine-learning-algorithms/ (accessed Jul. 27, 2021).

[11] S. Frost, B. Tor, R. Agrawal, and A. G. Forbes, “CompostNet: An Image Classifier for Meal Waste,” presented at the 2019 IEEE Global Humanitarian Technology Conference, GHTC 2019, 2019. doi: 10.1109/GHTC46095.2019.9033130.

[12] D. O. Melinte, D. Dumitriu, M. Mărgăritescu, and P.-N. Ancuţa, “Deep Learning Computer Vision for Sorting and Size Determination of Municipal Waste,” *Lecture Notes in Networks and Systems*, vol. 85, pp. 142–152, 2020, doi: 10.1007/978-3-030-26991-3_14.

[13] J. Liu and Y. Jiang, “Design of intelligent trash can be based on machine vision,” 2020, vol. 11584. doi: 10.1117/12.2579291.

[14] Y. Liu, S. Zhong, Z. Tian, and K. He, “Design of Vision Servo Sorting Robot System Based on SVM,” *J. Phys.: Conf. Ser.*, vol. 1550, p. 022032, May 2020, doi: 10.1088/1742-6596/1550/2/022032.

[15] A. Hevner and Chatterjee, *Design Research in Information Systems: Theory and Practice*, vol. 22. 2010. doi: 10.1007/978-1-4419-5653-8.

[16] M. Valente, H. Silva, J. Caldeira, V. Soares, and P. Gaspar, “Detection of Waste Containers Using Computer Vision,” *Applied System Innovation*, vol. 2, p. 11, Mar. 2019, doi: 10.3390/asi2010011.

[17] H. Wang, “Garbage recognition and classification system based on convolutional neural network vgg16,” 2020, pp. 252–255. doi: 10.1109/AEMCSE50948.2020.00061.

[18] G. Thung, *garythung/trashnet*. 2021. Accessed: Jul. 27, 2021. [Online]. Available: https://github.com/garythung/trashnet

[19] M. O. Rahman, A. Hussain, E. Scavino, H. Basri, and M. A. Hannan, “Intelligent computer vision system for segregating recyclable waste papers,” *Expert Systems with Applications*, vol. 38, no. 8, pp. 10398–10407, 2011, doi: 10.1016/j.eswa.2011.02.112.

[20] A. Salmador, J. Cid, and I. Novelle, “Intelligent Garbage Classifier,” *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 1, pp. 31–36, Dec. 2008.

[21] A. Torres Garcia, O. Rodea-Aragón, O. Longoria-Gandara, F. Sánchez-García, and L. González-Jiménez, “Intelligent Waste Separator,” *Computacion y Sistemas*, vol. 19, pp. 487–500, Sep. 2015, doi: 10.13053/CyS-19-3-2254.

[22] Y. Chu, C. Huang, X. Xie, B. Tan, S. Kamal, and X. Xiong, “Recycling,” *Computational Intelligence and Neuroscience*, vol. 2018, pp. 1–9, Nov. 2018, doi: 10.1155/2018/5060857.

[23] L.-G. Omar, R.-A. Oscar, T.-G. Andres, and S.-G. Francisco, “Multimedia inorganic waste separator,” in *2013 IEEE International Conference on Multimedia and Expo Workshops (ICMEW)*, Jul. 2013, pp. 1–4. doi: 10.1109/ICMEW.2013.6618314.

[24] W.-L. Mao, W.-C. Chen, C.-T. Wang, and Y.-H. Lin, “Recycling waste classification using optimized convolutional neural network,” *Resources, Conservation and Recycling*, vol. 164, 2021, doi: 10.1016/j.resconrec.2020.105132.

[25] C. J. Baby, H. Singh, A. Srivastava, R. Dhawan, and P. Mahalakshmi, “Smart bin: An intelligent waste alert and prediction system using machine learning approach,” in *2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, Mar. 2017, pp. 771–774. doi: 10.1109/WiSPNET.2017.8299865.

[26] S. N., P. M. Fathimal, R. R., and K. Prakash, “Smart Garbage Segregation amp; Management System Using Internet of Things(IoT) amp; Machine learning(ML),” in *2019 1st International Conference on Innovations in Information and Communication Technology (ICIICT)*, Apr. 2019, pp. 1–6. doi: 10.1109/ICIICT1.2019.8741443.

[27] M. Arebey, M. A. Hannan, R. A. Begum, and H. Basri, “Solid waste bin level detection using gray level co-occurrence matrix feature extraction approach,” *Journal of Environmental Management*, vol. 104, pp. 9–18, Aug. 2012, doi: 10.1016/j.jenvman.2012.03.035.

[28] G. Mittal, K. B. Yagnik, M. Garg, and N. C. Krishnan, “SpotGarbage: Smartphone app to detect garbage using deep learning,” 2016, pp. 940–945. doi: 10.1145/2971648.2971731.

[29] J. E. T. Akinsola, “Supervised Machine learning Algorithms: Classification and Comparison,” International Journal of Computer Trends and Technology (IJCTT*)*, vol. 48, pp. 128–138, Jun. 2017, doi: 10.14445/22312803/IJCTT-V48P126.

[30] R. Briner and D. Denyer, “Systematic Review and Evidence Synthesis as a Practice and Scholarship Tool,” in Handbook of evidence-based management: Companies, classrooms and research, 2012, pp. 112–129. doi: 10.1093/oxfordhb/9780199763986.013.0007.

[31] A. Liberati *et al.*, “The PRISMA Statement for Reporting Systematic Reviews and Meta-Analyses of Studies That Evaluate Health Care Interventions: Explanation and Elaboration,” *PLOS Medicine*, vol. 6, no. 7, p. e1000100, Jul. 2009, doi: 10.1371/journal.pmed.1000100.

[32] I. Salimi, B. S. Bayu Dewantara, and I. K. Wibowo, “Visual-based trash detection and classification system for smart trash bin robot,” in *2018 International Electronics Symposium on Knowledge Creation and Intelligent Computing (IES-KCIC)*, Oct. 2018, pp. 378–383. doi: 10.1109/KCIC.2018.8628499.

[33] S. Shalev-Shwartz and S. Ben-David, *Understanding Machine learning: From Theory to Algorithms*. Cambridge: Cambridge University Press, 2014. doi: 10.1017/CBO9781107298019.

[34] S. Shalev-Shwartz and S. Ben-David, *Understanding Machine learning: From Theory to Algorithms*. Cambridge: Cambridge University Press, 2014. doi: 10.1017/CBO9781107298019.

[35] F. Heinrichs, “Using CRISP-DM to Grow as Data Scientist,” *Medium*, Dec. 03, 2020. https://towardsdatascience.com/using-crisp-dm-to-grow-as-data-scientist-a07ce3fd9d56 (accessed Sep. 01, 2021).

[36] M. Hassaballah and A. I. Awad, *Deep Learning in Computer Vision: Principles and Applications*. 2020. doi: 10.1201/9781351003827.

[37] L. Bottou, “Large-Scale Machine learning with Stochastic Gradient Descent,” *undefined*, 2010, Accessed: Sep. 01, 2021. [Online]. Available: https://www.semanticscholar.org/paper/Large-Scale-Machine-Learning-with-Stochastic-Bottou/fbc6562814e08e416e28a268ce7beeaa3d0708c8

[38] K. Gurney, “Introduction to Neural Networks.” Taylor & Francis, Oxford.

[39] D. P. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization,” *ICLR*, 2015.

[40] U. Ruby and V. Yendapalli, “Binary cross entropy with deep learning technique for Image classification,” International Journal of Advanced Trends in Computer Science and Engineering, vol. 9, Oct. 2020, doi: 10.30534/ijatcse/2020/175942020.

[41] C. Gonçalves, “Identificação Automática de Plantas Invasoras em Imagens Aéreas”

Master thesis, Telecommunications and Computer Eng., ISCTE-IUL, Lisbon, 2019.

[42] D. Powers, “Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation,” Mach. Learn. Technol., vol. 2, Jan. 2008.