Automated Recycling System utilizing the KUKA youBot

Michael Verdon, *University of Portsmouth, IEEE*

Abstract— Climate change has profound effects on the planet and causes irreversible damage, with extreme weather potentially endangering humans and wildlife [1]. One countermeasure to climate change is to recycle more of what we use, but this can be very labour Intensive and expensive. This report explores the use of robotics to improve recycling, using the KUKA YouBot within a simulated environment (V-Rep). This report explores the challenges of recycling, how the recycling process can be optimised with the use of robotics, and the benefits of applying robotics to the problems of recycling.

1 Introduction

1.1 Why Recycle?

THE use of recycling is a poweful method to reduce the emission of greenhouse gasses. Over 90 percent of greenhouse gas emissions come from methane in landfills and wastewater alone, which is 18 percent of human-caused ethane emissions and 3 percent of total greenhouse gas global emissions [2]. As a result, leaving a social impact making people conscious of what they are buying and if it is recyclable, inspiring companies to consider what they are packaging their products with.

1.2 How Recycling Works

The recycling process begins with collection at kerbside from households, or other services based on what kind of location you live in (rural or urban). After which, recycling can be sorted by one of three common procedures: collection lorry at the kerb performing a 'kerbside sort', a 'two-stream' collection that only separates paper and card from cans, plastic bottles and jars in different compartments; and 'Co-Mingled', where waste is taken to a Materials Recovery Facility for sorting [3]. It Is this last method which I have selected as the basis for this project because large scale recycling sorting is a thorough process that cannot be done by just humans.

1.3 Materials Recovery Facilities

When materials are transported to a Materials Recovery Facility, a range of technology is used is used to sort materials (i.e., Rotating trommels to sort items by sizes and eddy current seperators for steel and aluminium [4].) by their inherent physical properties through a series of mechanical and manual processes to ensure maximum yield of useable recycling material [4]. The manual processes can introduce danger to the employees of such facilities such as exposure to airborne pathogens. Furthermore, conveyer belts are used to mitigate the impact of constant heavy lifting and repeated movements that can lead to musculoskeletal injuries and many more risks [4]. Working at these facilities is a hazardous job that requires careful risk assessment and safety measures to ensure human safety.
Another issue that arises in these recovery facilities is the contamination of unrecyclable waste products such as food waste. In Leicestershire, England alone, last year 5,500 tonnes of recycling material had to be discarded due to contaminants, costing almost £500,000. [5].

The use of robotics could address the danger issues of recycling by taking human labour out of it and potentially lead to a higher yield of used recycled material.

2 Previous Research

2.1 MIT and Yale's RoCycle System

MIT and Yale's 'RoCycle' robot uses a gripper that is resistant to puncturing, as waste can be sharp, through the use of handed shearing auxetics (HSAs) for the actuator basis. This metamaterial allows for directional actuation from a simple motor due to its strong geometric structure [7].

The gripper determines which type of item is being held through its deformation under pressure and uses a sorting algorithm to identify and correctly sort each item [6].

Among 14 items tested with their developed classifier uses the 'mean' and 'variance' of item deformations, the robot correctly identified 85% of the items which looks promising. Objects that were problematic were ones that had a mixture of materials on them such as a metal tin with paper on it, which was then identified as paper. Other classification techniques such as k-nearest neighbours yielded an accuracy of 23% which is not viable [6].

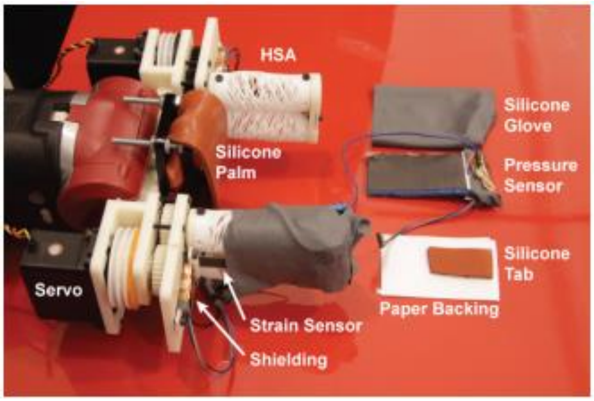

Figure 1: Structure of the gripper
Source: [6]

2.2 The Nail and Screw Recycling Robot

Computer vision technology and neural networks were used in a robot similar to the KUKA YouBot in order to identify nails and screws specifically for recycling within a construction site. This robot is comprised of a motion unit, a robot arm unit (with 5 degrees of freedom) and two detection units [8]. Due to the uneven and complex surfaces of a construction site, the robot is designed like a crawler robot which are shown to navigate more complicated terrain with the ability to climb obstacles and go over slopes [9].

Each sub-area of a construction site area is represented by a node within the neural network to scan through each area with path planning implementation. Increasing the number of nodes within the neural network will lead to an increase in computational complexity so each node consists of a fairly large area to maximise algorithmic efficiency [8].

As for individual item detection, a ‘Fast R-CNN’ framework is used which utilises an ROI Pooling Layer to extract vectors of the same length from each region of interest proposed being quicker than the classic selective searching algorithms utilised for generating region proposals [10].

Figure 2: The Nail and Screw Recycling Robot
Source: [8]

3 Methodology

3.1 The Setup

Similar to the Nail and Screw the Recycling robot, the KUKA YouBot is a mobile robot arm with 5 degrees of freedom and four mecanum wheels that allow for omnidirectional movement and a linear gripper. On the back of this robot is a tray to enable temporary item storage. The model of this robot is readily available within the software V-Rep (Copelia Sim).

Attached to the robot are four proximity sensors in all directions and a low-resolution orthographic vision sensor on the front.

The four proximity sensors allow for obstacle detection in all 4 directions, as the robot can move in all directions without needing to turn and allow for safe robot and human interaction.

A low-resolution orthographic vision sensor will be used to analyse the materials through colour in this example store within a cache to enable item sorting. Making the camera low-resolution will allow for quicker and easier image processing to determine which material is being picked up.

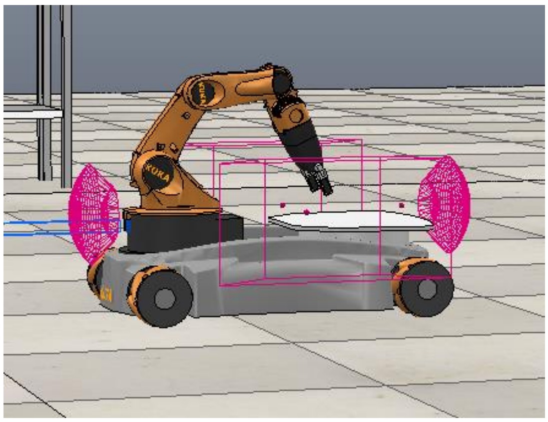

Figure 3: The KUKA YouBot with attached sensors and their ranges.

3.2 The Environment

For our experiment, the robot operated on a flat surface as it will be used in a recycling facility. Objects will be presented to the robot through a conveyer system which will halt upon detection at the end of the belt and a waste conveyer that will transport waste upon detection.

Radio beacons will be orientated in key locations around the facility such as the conveyer belt pickup location, the drop locations and a default location which will act as a key point between each path the robot takes. The use of such beacons aims to reduce time from the robot looking around to seek out the destination and reduce computational power needed to perform each sorting cycle.

Cubes will be spawned on a conveyer belt with the same coordinates to depict a funnelling system to ensure uniform positions of each material to be picked up. They will be either coloured red, blue, green or black for waste.

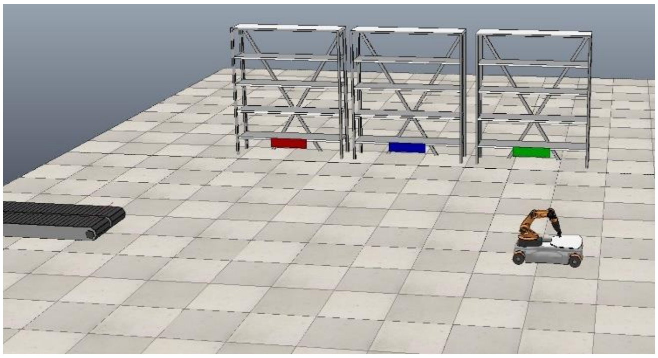

Figure 4: The environment

3.3 Behaviour

In the beginning of each sequence, the robot will orientate to the centre position to commence then drive towards the conveyer belt for item sorting. The vision sensor will identify whether the object will be picked up or discarded. After each recyclable material is picked up, the type of the material (denoted by colour) will be stored in a cache within the robot's memory. After two items are stored on the tray and one item is stored within the gripper, it will then reorientate to each position to drop the materials off. Everytime waste material is identified, it will not store in the array and continue with the picking process.

As the robot is carrying the materials, it orientates the gripper behind the robot to be less hazardous. Furthermore, the proximity sensors have short range meaning if the robot arm was facing forwards, the robot could collide with a human before the proximity sensor is triggered.

When the robot reaches the centre position, it will look at the last element of the array and drive to the designated location for the material and drop it off then repeat. This continues until the cache is empty to which another cycle is started.

3.4 Algorithms

All Algorithms are written in the scripting language 'lua' utilising the CopeliaSim API to interact with the software. Some details of the following presentations of the algorithms were omitted or altered for simplicity.

Main Sequence:

```
Sort Sequence:
--------------------------------------------------------------------------------
function sequence():
        goToCentre()
        while (cacheSize < 3) do
                pickupBoxFromPlace(pickupPositionBeacon)
        //Each pickup sequence reads vision sensor before pickup
        end

        for i=1,3 do
                goToCentre()
                dropPosition=decideDropPosition(4-i)
                //Goes from last element of array each time
                //As the last box picked up will be stored in gripper
                if (cacheSize == 2) then
                        pickupFromPlatform(storageSlot2)
                elseif (cacheSize == 1) then
                        pickupFromPlatform(storageSlot1)
                end
                cacheSize = cacheSize -1
                table.remove(cache, 4-i)
                //Remove last element in array
                //after each drop
                dropToPlace(dropPosition)
        end
end
--------------------------------------------------------------------------------
```

Figure 5: Main algorithm

This algorithm is the the brain of the robot. A sequence is defined as picking up all of the test cubes and dropping them to their respective locations whilst disposing of trash. This sequence will loop until the facility closes. Each time the robot travels to a key location, it positions itself far enough to avoid the proximity sensors having a positive reading but close enough to where the gripper will have more than enough space to pickup the objects.

Item pickup:

```
Pickup Object:
--------------------------------------------------------------------------------
function pickupBoxFromPlace():

        matrix,angle = getPositionAndAngleFromRadioBeacon(pickupPositionBeacon)
        //For use in driving the robot to its target destination.

        driveRobotToPosition()
        dispose=checkVisionSensor()
        //Returns true if cube is trash and false if the cube is needed.

        moveJointsToPickupPosition()
        pickupObject()

        if (dispose) then
                disposeOfCube()
        end
        //Disposes of cube if dispose returns true.

        if (cacheSize == 1) then
                dropToStorageTray(storageSlot1)
        elseif (cacheSize == 2) then
                dropToStorageTray(storageSlot2)
        end

end
--------------------------------------------------------------------------------
```

Figure 6: Item pickup and disposing algorithm

During the execution of this, the function calls the vision sensor to check what object we have and store it in a cache, if we wish to pick it up and sort or dispose of it if it is waste.

Proximity Sensor:

```
Proximity Sensor:
--------------------------------------------------------------------------------
if (frontSensor or rearSensor detects) then
        SetWheelVelocities(0)
end

if (leftSensor or rightSensor detects) then
        SetWheelVelocities(0)
end
--------------------------------------------------------------------------------
```

Figure 7: Proximity sensor algorithm

Proximity sensor checking is embedded within the vehicle control code, which will turn the wheels off as long as something is detected, to avoid collision with humans or obstacles, enabling safe robot-human interaction.

Vision Sensor:

```
Vision Sensor:
--------------------------------------------------------------------------------
function checkVisionSensor():
        image = getVisionSensorImage()
        //Take an image
        redPixel = image:byte(1)
        bluePixel = image:byte(2)
        greenPixel = image:byte(3)
        //Store rgb values

        if(150 < redPixel < 255) then
                table.insert(cache,"Red")
                return false
        elseif(150 < greenPixel < 255) then
                table.insert(cache,"Green")
                return false
        elseif(150 < bluePixel < 255) then
                table.insert(cache,"Blue")
                return false
        else
                return true
                //returns true for any unrecognized material
                //works specifically for red green and blue but colours
                //in between could trigger the other if statements.
        end
end
--------------------------------------------------------------------------------
```

Figure 8: Vision sensor algorithm

Boolean values are returned to determine whether the cube gets disposed of or colour is stored in cache and moved to the storage tray. An image is taken and the RGB values are extracted for analysing.

Each image of a cube taken will be simple to analyse due to the low-resolution camera making image analysis efficient.

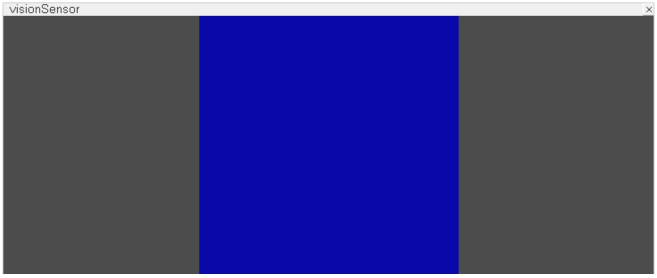

Figure 9: Low-Resolution Vision Sensor observing a blue cube.

4 Results Analysis and Evaluation

4.1 Main Sequence

Numerous iterations of the main sequence were executed to see whether the robot performs consistently and if there are any issues with picking up or dropping items.

Sequence No.	Cubes Picked Up	Waste Successfully Disposed (%)	Successful Cube Drops (%)
1	{Blue,Green,Blue}	100%	100%
2	{Blue,Blue,Red}	100%	100%
3	{Red,Red,Red}	100%	33%
4	{Blue,Red,Blue}	100%	67%
5	{Green,Red,Blue}	100%	67%
6	{Red,Blue,Green}	100%	100%
	Success Rate (%):	100%	78%

Figure 10: Main sequence experiment results.

Overall, the system was 78% effective at getting the cubes to reach their destinations. Nearly all of the failure occurred when the robot was trying to pick up from storage slot two. Every cube that reached a destination went to the correct shelf.

4.2 Vision Sensor Readings

The colours detected by the vision sensor from each main sequence were logged to see what is being picked up and the RGB values, and whether the cache behaves appropi-ately.

Cube No.	Colour	Vision Sensor Output	Red Value (0-255)	Green Value (0-255)	Blue Value (0-255)
1	Blue	Blue	12	12	173
2	Blue	Blue	12	12	173
3	Red	Red	172	12	12
4	Black	Black	12	12	12
5	Green	Green	12	172	12
6	Red	Red	172	12	12
	Success Rate (%):	100%			

Figure 11: Vision Sensor Test Results.

Every item was succesfully logged as the right colour and black cubes not being logged. The vision sensor behaved perfectly.

4.3 Proximity Sensors

A human dummy stood in numerous positions around the facility in the robots planned path to test the effectiveness of the proximity system. As expected the robot had been problematic when it reorientated from the centre to the drop positions as the robot rotates meaning the proximity sensors do not get triggered. This was succesful if the robot was going directly forwards, sideways or backwards.

5 Issues

5.1 Vision Sensor

The RGB values can vary depending on the lighting of the facility, this will impact readings and possibly lead to errors in colour evaluation.

In addition, materials coming in realistically will vary a lot more in colours and a different system should be used to detect material type, possibly the use of Computer Vision or something similar to pressure testing with a soft hand gripper [6].

5.2 Picking Up and Dropping

Errors in picking up occurred in retrieval from platforms, because the picking from platform system relies on fixed joint configurations and the material could slip and slide around the tray during movement.

Preventative measures such as localisation from visual input from the tray and high friction material on the tray could eradicate this problem.

5.3 Proximity Sensor Blind Spots

The KUKA YouBot can move in all directions without needing to rotate so placing a forward-facing proximity sensor alone is not enough. Placing four different sensors around made the robot safer for human-robot interaction but has blind spots, the robot could be made to face the direction it is heading to make the use of one proximity sensor effective or more can be built into the robot.

6 Conclusions

I have shown a mobile gripper arm robot can be used to sort recyclable material out with a promising success rate. In order to make this system fully viable for a Materials Recovery Facility, better adoption of proximity sensors should be used, with the possibility of using libraries such as OpenCV to analyse objects based on shape aswell or something else and not just colour, which is not sufficiently viable on Its own.

Moreover, the system for picking up items from storage must be strengthened, as materials must not be dropped or left out. This could prove hazardous if said materials are glass or sharp metallic parts, and additional safety procedures might be needed.

Potentially the use of multiple robots in a sequence could be used to speed up the process, with all these improvements put into place. Another robot could be used to pick up waste so the mobile robot will not need to consider waste material.

Such a system could potentially lead to efficient recycling and sorting, reducing the danger to humans through manual sorting, and increase the rate of work done. With further development, perhaps in the future, we can look forward to mobile robots deployed on a large commercial scale to sort out recycling in neighbourhoods, without needing to do it at a facility.

7 Acknowledgment

This work was completed with the support of the University of Portsmouth providing an IT facility to research this. The author would like to thank the university for access to its facilities.

8 References

[1] The effects of climate change (2022) NASA. NASA. Available at: https://climate.nasa.gov/effects/ (Accessed: January 26, 2023).

[2] Chertow, M. (no date) The ecology of Recycling, United Nations. United Nations. Available at: https://www.un.org/en/chronicle/article/ecology-recycling (Accessed: January 26, 2023).

[3] What happens to my recycling? (no date) Recycle Now. Available at: https://www.recyclenow.com/how-to-recycle/what-happens-to-my-recycling (Accessed: January 29, 2023).

[4] Materials Recovery Facilities (mrfs) (no date) Ergonomics at materials recovery facilities. Available at: https://www.hse.gov.uk/waste/mrf.htm (Accessed: January 29, 2023).

[5] Recycle contamination in recycling (no date) Less Waste. Available at: https://www.lesswaste.org.uk/recycle/contamination-in-recycling/ (Accessed: January 29, 2023).

[6] L. Chin, J. Lipton, M. C. Yuen, R. Kramer-Bottiglio and D. Rus, "Automated Recycling Separation Enabled by Soft Robotic Material Classification," 2019 2nd IEEE International Conference on Soft Robotics (RoboSoft), Seoul, Korea (South), 2019, pp. 102-107, doi: 10.1109/ROBOSOFT.2019.8722747.

[7] Chin, L. Handed shearing auxetics, Handed Shearing Auxetics | Lilly Chin. Available at: https://lillych.in/research/robotics/hsa.html (Accessed: January 29, 2023).

[8] Z. Wang, H. Li, X. Zhang, Automation in Construction Volume 97, 2019, Hong Kong, p. 220-228. (Accessed: January 30, 2023).

[9] W. Wang, Z. Du and L. Sun, Dynamic Load Effect on Tracked Robot Obstacle Performance, 2007 IEEE International Conference on Mechatronics, Kumamoto, Japan, 2007, pp. 1-6, doi: 10.1109/ICMECH.2007.4280041.

[10] R. Girshick, Fast R-CMN, International Conference on Computer Vision ICCV, 2015, (Accessed January 30 2023).