Intelligent prototype for the collection of small urban solid waste

Prototipo inteligente para la recolección de residuos sólidos urbanos de tamaño pequeño

ROSADO-CASTELLANOS, Damián Uriel†, PACHECO-FARFÁN, Ivette Stephany, FUENTES-CHAB, Iván Humberto and CHABLE-JIMENEZ, Eliezer

Instituto Tecnológico Superior de Escárcega. Calle 85 s / n between 10B.Col. Unidad, Esfuerzo y Trabajo, No. 1, Mexico.

ID 1st Author: Damian Uriel, Rosado-Castellanos / ORC ID: 000-0002-5344-1577, CVU CONACYT ID: 814618

ID 1st Co-author: Ivette Stephany, Pacheco-Farfán / ORC ID: 0000-0001-5405-0335, CVU CONACYT ID: 616907

ID 2nd Co-author: Iván Humberto, Fuentes-Chab / ORC ID: 0000-0001-9902-9011, CVU CONACYT ID: 856345

ID 3rd Co-author: Eliezer, Chable-Jimenez

DOI: 10.35429/JTP.2021.19.7.1.7

Received January 10, 2021; Accepted June 30, 2021

Abstract

Wasting disposal systems are a crucial sector in society, improper management of solid urban waste (SUW) can be harmful to health and the environment. In the state of Campeche, Mexico, an average of 3528.32 KG of waste is collected by every person, around 227 people are hired for this activity. This means that the generation of waste exceeds the collecting capacity in the state. This work presents an intelligent prototype for the collecting of small SUW. To achieve this task, we designed and created a model based on different electronic components (Raspberry, sensors, camera, etc.), as well as we created and implemented a SUW collection algorithm for its autonomy. The design of the prototype consists of two metal structures and the electronic components are integrated into the Raspberry. The creation and implementation of the algorithm work with artificial

Prototypes, Urban Solid Waste Management, Intelligent Systems, Artificial Vision

Resumen

Los sistemas de eliminación de desechos son un sector crucial en la sociedad, un manejo inadecuado de los residuos sólidos urbanos (RSU) puede ser perjudicial para la salud y el medio ambiente. En el estado de Campeche, México se estima una media 3528.32 KG de desechos recolectados por persona de un total de 227 personas contratadas para esta actividad. Esto indica que la generación de desechos sobrepasa la capacidad de recolección en el estado. En este trabajo se presenta un prototipo inteligente para la recolección de RSU de tamaño pequeño. Para lograr esta tarea diseñamos y creamos un modelo basado en diferentes componentes electrónicos (Raspberry, sensores, cámara, etc.), así como creamos e implementamos un algoritmo de recolección de RSU para su autonomía. Los resultados muestran el proceso de desarrollo del prototipo, así como el código generado en el lenguaje de programación Python.

Prototipos, Manejo de Residuos Sólidos Urbanos, Sistemas Inteligentes, Visión Artificial

Citation: ROSADO-CASTELLANOS, Damián Uriel, PACHECO-FARFÁN, Ivette Stephany, FUENTES-CHAB, Iván Humberto and CHABLE-JIMENEZ, Eliezer. Intelligent prototype for the collection of small urban solid waste. Journal of Technological Prototypes. 2021. 7-19:1-7.

† Researcher contributing first author

Introduction

Throughout history, human activity has always generated solid urban waste (MSW). MSW is a heterogeneous material with different production rates and composition depending on the place and the season (Qdais H. et Al, 1997). Waste disposal systems are a crucial sector in human civilization (Hossain S. et Al, 2019), which can be harmful to health and the environment without proper management. Only in Campeche, Mexico the generation of MSW amounts to 800,930 Kilograms per day of which there are 227 people for the collection at the state level according to the study of (Ministry of Urban Development and Environment, SF) giving an average 3528.32 Kilograms per person, which indicates an excessive workload making it difficult to cover MSW collection in the state.

Waste disposal systems require the implementation of MSW collection automation. In different works of the state of the art, different solutions are proposed based on the creation of remote-controlled and autonomous robots with artificial vision techniques. In (Mehta N. et Al, 2018) they propose a solution based on the collection of MSW with Arduino, GPS and different elements, whereby means of Google Earth coordinates it recognizes the areas to be cleaned. In (Samano Villegas R.E et Al, 2016) a remote-control prototype based on Arduino and other electrical components for the collection of objects is proposed. (Shamim Hasan A. et Al, 2017) proposes a solution of an object collector using video transmission and remote control based on smart mobile devices based on Arduino and the Android operating system. In (Gourav S. et Al, 2017) a MSW collector is proposed for domestic homes and work offices, its proposed methodology is based on an intelligent algorithm to collect without the need for human intervention. (Almosalami A. et Al, 2018) propose a system for collecting MSW on beaches through a network of interconnected and tele-operated robots in a virtual reality environment. In (Khandare S. et Al, 2018) he proposes a robot to clean the MSW around a trash can based on image processing and an ultrasonic sensor.

(Hossain S. et Al 2019) proposes an autonomous MSW collector using a deep learning algorithm on a Raspberry Pi and electronic devices. In (Othman H. et Al, 2020) they propose the common elements, methodology and classification of cleaning systems as a basis for new collectors.

This article proposes the design and implementation of a RSU collector prototype based on artificial vision techniques, *Raspberry Pi* and other electronic components. The main objective is that autonomously and without human intervention perform a 360-degree search in the environment to locate objects classified as MSW and carry out the collection.

Methodology

For the development of the RSU collector prototype, see figure 1, a digital sketch developed in the Free version of the SketchUp design and modeling software is considered as the basis, which consists of two levels, plus a protective casing. The first level is intended to store the battery, travel and harvest servo motors. The second level stores the Raspberry Pi, the breadboard, and the connections to different electronic components. Finally, the protective casing allows the collector prototype to be turned on / off, as well as to contain the video screen so that the user can view the images of the robot's camera and its operation.

Figure 1 Digital sketch of the prototype. The Figure on the left is the first level of the prototype where battery and servo motors are integrated. The Figure on the right is the second level that contains the *Raspeberry Pi*, breadboard and connections
Source: (Own elaboration)

The integration of the electronic components considers different devices, see table 1, which allows the operation of the collection algorithm to search, locate and collect MSW.

Component	Description
Breadboard	600 perforations with dimensions of 6.4 cm front by 8.6 cm high.
Set of cables	Dupont Type Cables. 40 plug-to-plug cables. 40 jacks to plug cables. 28AWG gauge. Length 15 cm.
Ultrasonic sensor	Tx and Rx sensor in one. Detection range 1.7 to 400 cm. 40kHz frequency. 15 degree opening angle and 5V power supply.
Servo motors	Servo motor with a 6.5 to 12 Kgf / cm stem with 360-degree rotation and 4.8-6V input with 100mA maximum. Weight of 55 grams with dimensions of 40 by 20 by 39 mm.
Raspberry Pi	Model 3B with 4GB of RAM.
Camera	Pi Cam version 2 with 8MP.
Drums	Brand Tedge Model CP 10000
Screen	Generic display with 3.5-inch HDMI input.

Table 1 Electronic components
Source: (Own source through data collection)

The proposed algorithm implements artificial vision techniques such as the detection of objects by segmentation of color spaces, (Gil, P. et Al, 2004), which allow the detection of the position of the object and its classification as an RSU object or not, as well as the functionality of the electronic components integrated in the collector prototype. The stages of the algorithm are made up of 4 general steps for collecting MSW:

Search for MSW: In this stage the detection of MSW is evaluated based on the sequence of images obtained from the camera. If the object is not detected, the collecting robot will rotate on its own axis, otherwise it stops the servomotors that allow the rotation.

Align Proximity Sensor with RSU: In this stage, an alignment adjustment is made between the location of the proximity sensor and the position of the RSU detected in the image. This ensures that the measured distance corresponds to the RSU and not to another object.

Approach RSU: In this stage the distance of the RSU is evaluated based on the current position of the robot. If the distance is less than a defined threshold, it stops the servomotors that allow the approach, otherwise it will continue advancing.

Collect RSU: In this stage the collection of the object begins through the servomotors implemented in the robot gripper.

Figure 2 Object collection algorithm composed of 7 different stages for the collection of MSW
Source: (Own elaboration)

The workflow is carried out by using servo motors to move the robot on its same axis at a 360-degree angle. During the turn, the evaluation of a descriptor based on color spaces is made for the detection of MSW. The descriptor is generated from the conversion of the RGB color space obtained from the camera image to the HSV color space (Nallaperumal K. et Al, 2013), where the pixel values of the H channel are obtained, which represents the purity of the color of the MSW composition and from the definition of two thresholds it is determined whether the object of study is considered as MSW or not.

Once the RSU is found, the next step is to stop the servomotors so that the robot stops rotating on its own axis and makes a small turn in the opposite direction to align the proximity sensor with the detected RSU, this step is carried out due to a Image processing delay on the raspberry given the capabilities of the CPU.

As soon as the robot is aligned, the RSU object detection stops working to reduce the processing load and turn on the servomotors that allow movement based on the distance provided by the proximity sensor. If the distance is less than the threshold defined by 5 cm, the collector prototype will approach the object, otherwise it stops the measurement of the proximity sensor and the servomotors.

When the distance is ideal for the robot's grippers, the next step is to turn on the servomotors that allow the object to be collected.

Results

MSW collector prototype

The construction of the prototype starts from the definition of the connection circuit of the electronic components, see Figure 3, where the Protoboard will be providing a 5V supply, as well as a ground bridge to the components, while the direct connection to the Raspberry Pi allows communication for the execution of the components.

Figure 3 Connection diagram of electronic components connected to the *Raspberry Pi* and used in the creation of the RSU collector prototype
Source: (Own elaboration)

The connection diagram consists of a 10,000 mA battery which generates a 5V output to power the Raspberry Pi. An 8 MP camera is also integrated which is connected directly to the CAMERA port of the raspberry, in the same way a video screen is integrated into the HDMI port. The proximity sensor and the servomotors that allow movement are connected to the breadboard as a power supply in different rows, as well as the connection with the pins of the Raspberry.

The structure of the robot is based on the dimensions of the electrical components. Therefore, the first step was to measure and cut the base metal structure composed of two levels. The first level is made up of the displacement and harvesting servomotors, see Figure 4, where the assembly was carried out with 8cm support bars and 1-inch screws, as well as the wheels were integrated. For the harvesting servomotors, the clamp was integrated with metal extensions to allow a precision grip and a space was left for mounting the battery and connection cables.

Figure 4 Assembly of servomotors on the first level. The Figure on the left shows the basic structure and the integration of the servomotors. The Figure on the right shows the integration of the battery and the connection cables for the *Raspebberry Pi*
Source: (Own elaboration)

The second level consists of the Raspberry and other elements, see Figure 5, where the mounting was done with half-inch screws to the metal plate. Once assembled, the connection with the servomotors was made, as well as the integration of the proximity sensor in the lower front part and the camera in the upper front part without obstructing the collector clamp.

Figure 5 Assembly of Raspberry Pi and other electronic components on the second level. The Figure on the left shows the front view of the prototype integrated by the proximity sensor and servo motors for collecting objects. The Figure on the right shows the aerial view of the Raspberry mounting, camera and connections to the electronic components
Source: (Own elaboration)

Finally, to protect the circuit, see Figure 6, a metal casing was made to protect it from the surrounding environment. At the top it integrates the video screen and an on / off button that allows you to start the Raspberry Pi. At the bottom there are a series of insulators to avoid direct contact of the circuit with metal in order to protect the electronic components and avoid any short.

Figure 6 Circuit protective shell. The left Figure shows the exterior of the protective case and its power button. The Figure to the right shows the inside of the protective case and the connections of the power button with the Raspberry Pi
Source: (Own elaboration)

The integration of all these elements results in the MSW collector prototype, see Figure 7, given its design and structure, it allows different movements such as moving forward, backward and turning on its own axis, this helps a suitable movement for the search and collection of MSW. . Likewise, due to the electrical components used, the robot supports a maximum weight of 20Kg with a maximum width of 7cm. The base structure is mostly made of metal, particularly aluminum, painted blue for easy viewing in the environment in which it is located.

Figure 7 Object collector prototype. The Figure on the left is the final front view of the prototype integrated by camera, harvesting servomotors and proximity sensor. The Figure on the right is the lateral aerial view of the prototype showing the prototype's protective casing and scroll wheels
Source: (Own elaboration)

Once the prototype was completed, an algorithm created by the authors, see Figure 8, was implemented in the Python programming language and with the OpenCV computer vision library (Zelinsky, A., 2009) that provides the necessary autonomy to perform the task collection without human intervention.

```
stop = False
alineado = False
medirDistancia = True
recogerObjeto = False
alineado2 = False
while not stop:
    #, origen = cap.read()
    origen = cv2.resize(origen,size,interpolation=cv2.INTER_AREA)
    frame = cv2.medianBlur(origen,5)
    frame = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)
    h,s,v = cv2.split(frame)
    hs = cv2.merge((h,s))
    mask = cv2.inRange(hs,lower,upper)
    mask = cv2.erode(mask,kernel,iterations=2)
    mask = cv2.dilate(mask,kernel,iterations=5)
    #, contours, _ = cv2.findContours(mask, cv2.RETR_TREE, cv2.CHAIN_APPROX_SIMPLE)
    if len(contours) == 0 or len(contours) > 1:
        servomotorBuscar();
    if len(contours) == 1 and not alineado:
        area = cv2.contourArea(contours[0])
        if area > areamin:
            (x,y,w,h) = cv2.boundingRect(contours[0])
            m0 = ( int(x-(w/2)), int(y-(h/2)) )
            cv2.rectangle(origen,(x,y),(x+w,y+h),red,1,cv2.LINE_AA)
            cv2.rectangle(origen,(m0[0]-5, m0[1]-5),(m0[0]+5, m0[1]+5),white,1,cv2.LINE_AA)
            if(m0[0] > cenl[0] and m0[0] < cenr[0]):
                alineado = True
            else:
                servomotorBuscar();
        else:
            servomotorBuscar();
    if alineado and medirDistancia:
        distancia = sensorProximidad()
        if distancia > 11:
            if not alineado2:
                servomotorAlinear()
                alineado2 = True
                servomotorAcercar()
            else:
                medirDistancia = False
                recogerObjeto = True
        if alineado and recogerObjeto:
            servomotorRecoger()
            stop = True
            cv2.rectangle(origen,cenl,cenr,green,1,cv2.LINE_AA)
            cv2.imshow("frame",origen)
            cv2.waitKey(1)
            cap.release()
            cv2.destroyAllWindows()
```

Figure 8 RSU collector algorithm code developed in the Python programming language and with the OpenCV machine vision tool
Source: (Own elaboration)

The algorithm allows the manipulation of all electronic components for their proper operation from the sequence of images obtained from the camera. Given the image, if the object is not detected, the prototype will keep rotating on its axis. If the RSU is detected the prototype will stop rotating and will make an alignment with the RSU and the proximity sensor. Once aligned, the robot will stop processing the images and will move towards the object as long as the distance is greater than 5 cm, otherwise the robot will stop and collect the object.

The MSW collector prototype allows the automation of solid waste collection processes in both public and private areas, being a useful tool for improving the service of waste disposal systems in the state of Campeche. This means that the workload of the cleaning staff can be considerably reduced by automating the collection process. This allows better management of MSW, which reduces the probability of damage to human health and the environment.

Acknowledgments

This research project was carried out thanks to the support of the National Technological Campus Escárcega (Higher Technological Institute of Escárcega), which provided the necessary spaces for its development and execution.

Conclusions

The results obtained show that the proposed design with a simple and easy-to-navigate structure allows the integration and proper operation of the implemented algorithm and the electronic components connected to the Raspberry Pi, given that it has a limited amount of memory and processing.

The creation and implementation of the model from the metal structures and the integration of different electronic components such as Raspberry Pi, camera, proximity sensor, servo motors, among others used in the development methodology, allow the creation of a physical prototype for collecting MSW.

The algorithm proposed in the Python programming language and based on artificial vision techniques with the OpenCV tool, allows to give autonomy to the physical prototype, so that, for the operation of electronic devices, it does not require human intervention and can perform actions to starting from the conditions given in the environment.

The development of this prototype provides a tool to support the human being that allows reducing times and increasing the collection capacity of the waste systems of the state of Campeche, Mexico.

References

- Qdais, H. A., Hamoda, M., & Newham, J. (1997). Analysis of residential solid waste at generation sites. *Waste Management & Research*, 15(4), 395-406.
<https://doi.org/10.1006/wmre.1996.0095>
- Hossain, S., Debnath, B., Anika, A., Junaed-Al-Hossain, M., Biswas, S., & Shahnaz, C. (2019a). Autonomous Trash Collector Based on Object Detection Using Deep Neural Network. *TENCON 2019 - 2019 IEEE Region 10 Conference (TENCON)*, 1406-1410.
<https://doi.org/10.1109/tencon.2019.8929270>
- Secretaría de Desarrollo Urbano y Medio Ambiente. Gobierno del Estado de Campeche. (s. f.). Programa Estatal para la Prevención y Gestión Integral de los Residuos Sólidos Urbanos. Recuperado 21 de enero de 2021, de <http://apps.semabicc.campeche.gob.mx/wp-content/uploads/2019/07/Programa-Estatal-Campeche.pdf>
- Mehta, N., Verma, S., & Bhattacharjee, S. (2018). Automatic Garbage Collector Bot Using Arduino and GPS. *Lecture Notes on Data Engineering and Communications Technologies*, 273-280.
https://doi.org/10.1007/978-981-10-8339-6_30
- Samano Villegas, R. E., Zamacona López, C. E., & Alarcón Paredes, A. (2016, septiembre). Prototipo de robot recolector de objetos. *Tlamati Sabiduría*, 7(1).
<http://tlamati.uagro.mx/t7e1/83.pdf>
- Shamim Hasan, A. S. M., Hossain Jewel, M. K., Mostakim, M. N., Bhuiyan, N. H., Rahman, M. K., Hossain, S. D., & Hossain, M. K. (2017). Smartphone Controlled Spy Robot with Video Transmission and Object Collector. *International Journal of Engineering and Manufacturing*, 7(6), 50-58.
<https://doi.org/10.5815/ijem.2017.06.05>
- Gourav, Singh, S., Singh, A., Singh, B., Singh, J., & Kaur, H. (2017). Designing and Modeling of Automatic Garbage Collector. *IJSRST*, 3(7), 949-953.
<https://doi.org/10.13140/RG.2.2.27402.16329>
- Almosalami, A., Jones, A., Tipparach, S., Leier, K., & Peterson, R. (2018, March). Beachbot: Crowdsourcing garbage collection with amphibious robot network. *In Companion of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, 333-334.
<https://doi.org/10.1145/3173386.3177832>

Khandare, S., Badak, S., Sawant, Y., & Solkar, S. (2018). Object Detection Based Garbage Collection Robot (E-Swachh). *International Research Journal of Engineering and Technology (IRJET)*, 5(3), 3825-3828.

<https://www.irjet.net/archives/V5/i3/IRJET-V5I3894.pdf>

Othman, H., Iskandar Petra, M., Chandratilak De Silva, L., & Caesarendra, W. (2020). Automated trash collector design. *Journal of Physics: Conference Series*, 1444, 012040.

<https://doi.org/10.1088/1742-6596/1444/1/012040>

Gil, P., Torres, F., & Ortiz, F. G. (2004, septiembre). Detección de objetos por segmentación multinivel combinada de espacios de color. RUA. <http://hdl.handle.net/10045/2179>

Nallaperumal, K., Subramanyam, M., Subban, R., Perumalsamy, P., Durairaj, S., Devi, S. G., & Kumar, S. S. (2013). An analysis of suitable color space for visually plausible shadow-free scene reconstruction from single image. 2013 *IEEE International Conference on Computational Intelligence and Computing Research*, 2476-2480.

<https://doi.org/10.1109/iccic.2013.6724298>

Zelinsky, A. (2009). Learning OpenCV---Computer Vision with the OpenCV Library (Bradski, G.R. et al.; 2008) [On the Shelf]. *IEEE Robotics & Automation Magazine*, 16(3), 100.

<https://doi.org/10.1109/mra.2009.933612>.