A COMPUTER MODEL FOR DETECTING AIRPORT LUGGAGE'S DIMENSIONS USING LOW-COST DEPTH SENSORS

Vitor de Almeida Silva, Marcos Paulino Roriz Junior*, Michelle Carvalho Galvão da Silva Pinto Bandeira
Faculdade de Ciência e Tecnologia, Universidade Federal de Goiás

*** Corresponding author e-mail address**: marcosroriz@ufg.br

PAPER ID: SIT149

ABSTRACT

A factor that impacts airlines' resources is the verification of luggage's dimensions during the boarding process. To check conformity, companies usually rely on a human operator. To mitigate this issue, companies are investing in automatic self bag drop systems. This process introduces new technological challenges, since, in this scenario, the verification of the conformity of the luggage dimensions is delegated to the passenger, which can lead to errors. In addition, current solutions use specific computing devices, such as laser scanners, that are expressive in size and cost, which may require interventions in the airport infrastructure. In this sense, this work proposes a model to measure luggage through a low-cost depth sensor as an alternative to obtain and verify its dimension. To do so, we developed an algorithm that obtains a 3D point cloud of the luggage surface through a Microsoft Kinect V2 sensor. After that, we generate the minimum polygon that cover these points, a process called convex hull. By doing so, we are able to obtain the luggage's dimensions. To test our approach, we implemented our model in MATLAB and created a real-world prototype. The results indicate that the mean absolute error is 1.33cm, 1.90cm, and 0.52 cm for the width, length, and depth, respectively, which indicates that this technology has the potential to become an alternative technology to detect the luggage's dimensions.

Keywords: self bag drop, luggage dimensions, internet of things, computer vision.

1. INTRODUCTION

From 2004 to early 2020, the number of people using air transport reached 4.723 billion (Statista, 2021). In this scenario, the boarding time of passengers increased significantly. For instance, Ren et al. (2020) study indicate an increase from 22 to 40 minutes in boarding time for 200 passengers in 1990 and 2009 respectively. In addition, it is known that the boarding process is one of the most time-consuming factors at airports (Gao et al., 2018). These delays result in losses for passengers and companies (Qingji et al., 2018).

Within the boarding process, checking the dimensions of luggages is one of the factors that consume substantial time (Ronzani & Correia, 2015; Negri & Borille, 2017). Another factor is the congestion formed in the aisle between the seats. One of the reasons for this comes from the delay in storing luggage by passengers in the aircraft top compartments, an action that is influenced by their dimensions, quantity, and material (Ren et al., 2020).

To address this issue, airlines have invested in automated solutions, which are part of the technological advances provided by Industry 4.0 (Göçmen, 2021).

Among the solutions, the implementation of self-service terminals to speed up the boarding process stands out (Colby, 2019; Ren et al., 2020). This procedure requires technologies that allow passengers to check in their own luggage (self bag drop). This logic can be applied at the entrance, in the waiting area, or along the airport (Alsyouf et al., 2018). It optimizes services and improves time savings. However, the process of verifying the luggage's dimension falls to the passenger, which can generate errors due to the format and position of the luggage (Colby, 2019; Ren et al., 2020).

To aid the passenger, airlines have explored the use of computer vision equipment as a way to detect the luggage size (Anderson, 2019; Göçmen, 2021). The most common technology applied are based on laser sensors, which are capable of obtaining information from luggage in the form of point clouds (Gao & Liang, 2021). These points represent where the sensor interacted with the luggage in an R^3 space.

From this point cloud, it is possible to provide an accurate measurement of the luggage's dimension (Chan et al., 2018). However, these equipments can have significant costs. Furthermore, in addition to the software, they commonly require physical instrumentation and modification of the infrastructure for their installation, which increases the difficulty in deploying or moving it to other parts of the airport (Gao et al., 2018).

Parallel to that, alternative and low-cost sensing technologies have emerged as a way to analyze an object's size (Kuan et al., 2019). For instance, the Microsoft Kinect, a depth sensor developed initially for video games, has encountered applications in several other areas, including airports (Zennaro et al., 2015; Anderson, 2019). This device has been widely adopted due to its feature and significantly low price (around 200 dollars) (Chan et al., 2018).

However, there is a lack of study that analyzes the usage and limitations of this technology for measuring luggages. Hence, given this gap, in addition to the inflexibility of existing technologies, and their expressive prices, this paper proposes a model for detecting the luggage size using low-cost depth sensors. To do so, the method employs the Microsoft Kinect depth sensor and algorithms for analyzing the point clouds. To evaluate our method, we programmed the approach in MATLAB and built a real-world prototype. Our results indicate that our model is able to detect the luggage's dimension with a mean absolute error of 1.33cm, 1.90cm, and 0.52 cm for the width, length, and depth. This result indicates that the model has the potential to be used in airports.

The rest of this paper is structured as follows. First, Section 2 presents the fundamental concepts that underlie this work. Then, Section 3 presents and discusses the method, while Section 4 presents the experiment used to evaluate our approach. Finally, Section 5 states the conclusion and future work associated with the limitations of the proposed method.

2. FUNDAMENTAL CONCEPTS

To exemplify the challenges in obtaining the luggage's dimension and verifying its conformity, we review the boarding process and how current technology approaches work.

2.1. BOARDING PROCESS

The boarding process analyzed was based primarily on the rules of the Brazillian National Civil Aviation Agency (ANAC, 2022), but can be customized by airlines. We also considered the rules for individual airlines, such as Azul (2022) and Gol (2022).

Passengers' luggages can be classified into carry-on, checked, and special luggage. Carry-on luggage are those that the passenger carry to the aircraft. According to ANAC, the standard limits for these type of luggage is 55 cm, 35 cm, and 25 cm for length, width, and depth, respectively. For maximum weight, the limit is 10 kg.

On the other hand, checked luggages must be dispatched, usually with the airline staff at the check-in desk or in a self bag drop machine. According to ANAC, the standard limits for checked luggage are 55 cm, 80 cm, and 28 cm, for length, width, and depth, respectively. And for the weight, the limit is 23kg. In addition to these issues, wheels, handles and any accessories attached to the luggage are also accounted in the measurement.

Finally, special luggages are those that do not fit into previous categories. For example, a sport equipment or a musical instrument usually needs to be dispatched as special luggage. Since their shape varies significantly, they are usually handled manually by airlines staff. As such, we will not address them in this work.

2.2. METHODS TO EXTRACT OBJECT DIMENSIONS

This is the main body of the article. Use 1 to 4 sections (e.g., bibliographic review, methodology, results, conclusions) This chapter is an example of the various chapters that the authors can use to report their respective studies, including the study methodology, the analysis and the final results.

The first review result indicated a total of 167 works. After a careful analysis, we found 14 papers that focus on how to obtain object dimensions. Of these studies, only 3 were trying to detect the airport luggage dimensions. The others acted in otherd domains. Regarding the technology, we found out that 7 works used the Microsoft Kinect, while 4 used laser scanners, and 3 used binocular vision. We highlight that none of the works that tried to analyze the luggage dimension used the Microsoft Kinect.

Regardless of the type of the technology used to scan the objects, the works are generally based on computer vision techniques. According to Bhowmik & Appiah (2018), computer vision is the process of acquiring, analyzing, and processing video and images to make decisions.

In computer vision, an object's dimensions is usually computed through a point cloud. These data points represent the object's contact points with the sensor in an 3D space, as illustrated in Figure 1 (a) and (b). Subsequently, a point cloud is processed to reduce noise. After that, we can compute the object's dimension by discovering a three-dimensional polygon that covers the cloud, see Figure 1 (c).

This entire process commonly employs the convex hull algorithm, since it obtains the smallest polygon that wraps the captured object points (Ding et al., 2018; Gao et al., 2018). To exemplify this process, consider the following example. Given a set of points A, the convex hull seeks to find border points, generating a subset B, which in turn represents the smallest polygon that covers all points of A. This polygon can be used to reproduce the surface of objects or collect information from its dimensions (Ding et al., 2018). Figure 1 (d) illustrates an example of 3D convex polygons.

Concerning the pre-processing step, we highlight that it is possible to apply several techniques, such as subsampling, clustering, segmentation, and filtering. Subsampling reduces the density of points, reducing the

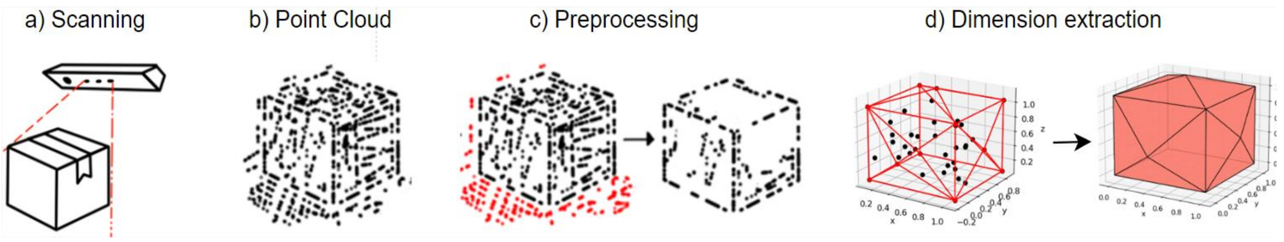


Figure 1 Process of obtaining dimensions using a depth sensor (Microsoft Kinect) and convex hull

computational expense needed to analyze the object (Ruchay et al., 2018). Clustering, on the other hand, removes outliers that can hinder the reconstruction process, since border points can significantly change the polygon that delimits the 3D object. Further, segmentation aims to remove outlier points that will not be used in processing (Limwattanapibool and Arch-int, 2017).

As for technologies for capturing point clouds, we highlight the usage of mobile laser scanners, binocular sensors, and depth sensors, such as Microsoft Kinect.

Laser scanners can be applied in self-service services. Such equipment is composed of a treadmill, an internal environment with lasers, and a processing center (Qingji et al., 2018). The customer places their luggage on the treadmill and the scanner, through the use of sensors, returns the dimension values (Gao et al., 2018).

Airline and airport usually employs this type of technoclogy to enable the self bag drop stations. However, laser scanners usually have a signifcant cost, ranging from USD 10,000.00 to USD 111,000.00 for a single sensor. As an airline can have multiple self bag drop stations, this can be costly. Furthermore, in addition to the software, they commonly require physical instrumentation and modification of the infrastructure for their installation, which can make it difficult or impossible to move them for other parts of the airport (Gao et al., 2018).

Another option is binocular vision systems. The technique is inspired by human vision, consisting of capturing two or more images of the same object at relatively different angles by cameras. From this, it is possible to form a 3D object and identify its dimensions (Qingji et al., 2018). The main disadvantage of this method is that the position of the luggage directly influences the result, sometimes making the method impractical (Gao & Yang, 2013).

As an alternative to the technologies mentioned, researchers have explored the use of low-cost depth sensors, such as Microsoft Kinect. This device is capable of efficiently returning point clouds with high precision (Chan et al., 2018). Another attraction is its portability, ease of installation, as well as its price, around USD 200 dollars (Ruchay et al., 2018). Figure 2 illustrates the Microsoft Kinect version 2 components.

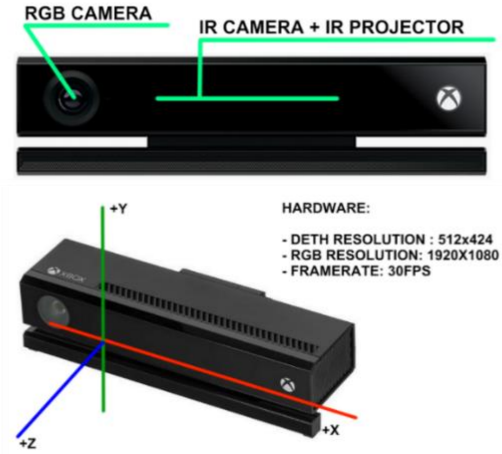


Figure 2 Microsoft Kinect V2

3. METHODOLOGY

The proposed model is based on two major phases: sampling strategy and dimension extraction. This section presents the concepts of each phase. Further, it describes the real-world prototype we built to implement the model

As stated previously, we decided to use the Microsoft Kinect v2 depth sensor due to its low cost and successful usage in other scenarios (Kuan et al., 2019). For implementing the algorithm, we used the MATLAB programming language due to its community support and compatibility with the sensing device. Finally, we used an Arduino board to control parts of the hardware device.

3.1. SENSING STRATEGY

The first step was to choose the sensing strategy. Precisely, how are we going to scan (sample) the luggage. Considering this task, we highlight two options: static sensing, and mobile sensing.

Static sensing consists of positioning the sensor at a fixed point and collecting the entire point cloud of the object in a single scan. This approach was used in works such as Gao & Yang (2013) and Qingji, Chuanbo & Qijun (2018), but without the use of Kinect. The resulting data sample can collect the point cloud in great detail. However, as the sizes of luggage can vary, it does not cover all cases, such as bags larger than the active area.

Moving on to mobile sensing, unlike the static mode, this approach does not scan the object in a single pass. In this case, the system sense a slice region, also known as Region Of Interest (ROI). The ROI is a segmentation that constrains the set of points in a region, as illustrated by the red slice in Figure 3. By using a sampling step and frequency, the object point cloud can be rebuilt.

The sensor is positioned at a fixed point, and the object is moved through its active region (ROI). Therefore, it is necessary to carry out a more extensive treatment consisting of collecting N samples of the object and concatenating them all in real-time to form the complete point cloud. Equation (1) shows the calculation, where p_i is the new sample taken and P_f is the resulting point cloud. The flowchart illustrated in Figure 6 represents this approach.

$$P_f = \cup_{i=0}^{n} p_i \quad (1)$$

The Kinect was positioned at 90° in relation to the luggage. This position reduces the number of occluded points, increasing the level of detail collected, such as handles and wheels. To fix the Microsoft Kinect, we built a support structure. The sensing device was placed at the top, in a glass support from a distance of 1 meter from the base.

A static sensor can be costly, as such, we decided to employ a mobile sensing strategy. Our approach uses the following steps (illustrated in Figure 7). First we choose the input parameters: the sampling rate (at which rate the point cloud will be captured), and the region of interest (location that will be sampled). After choosing the initial parameters, we check if there is an object between the treadmill and the sensor. If positive, we trigger the capture phase.

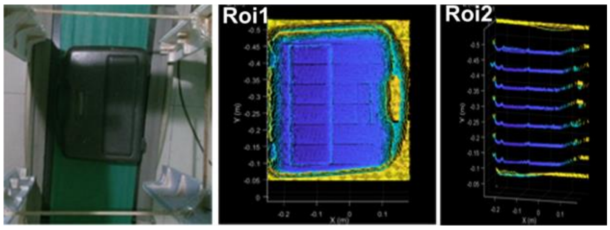


Figure 4. A luggage's point cloud. Roi with different sampling rate

In this phase, we capture the entire point cloud of the luggage. To avoid handling angled data points, we only keep the part of the point cloud that is located directly below the Kinect sensor (the ROI). For instance, consider Figure 4. A lower sampling frequency, such as 1 cm, means that every 1 cm the sensor will capture a point cloud and keep the points in the middle. When joining these points, we get the entire point cloud without distortion with a 1 cm precision. Notice the gap produced when using a different frequency, 5 cm. The holes represent the lack data of caused by the delay between each sample.

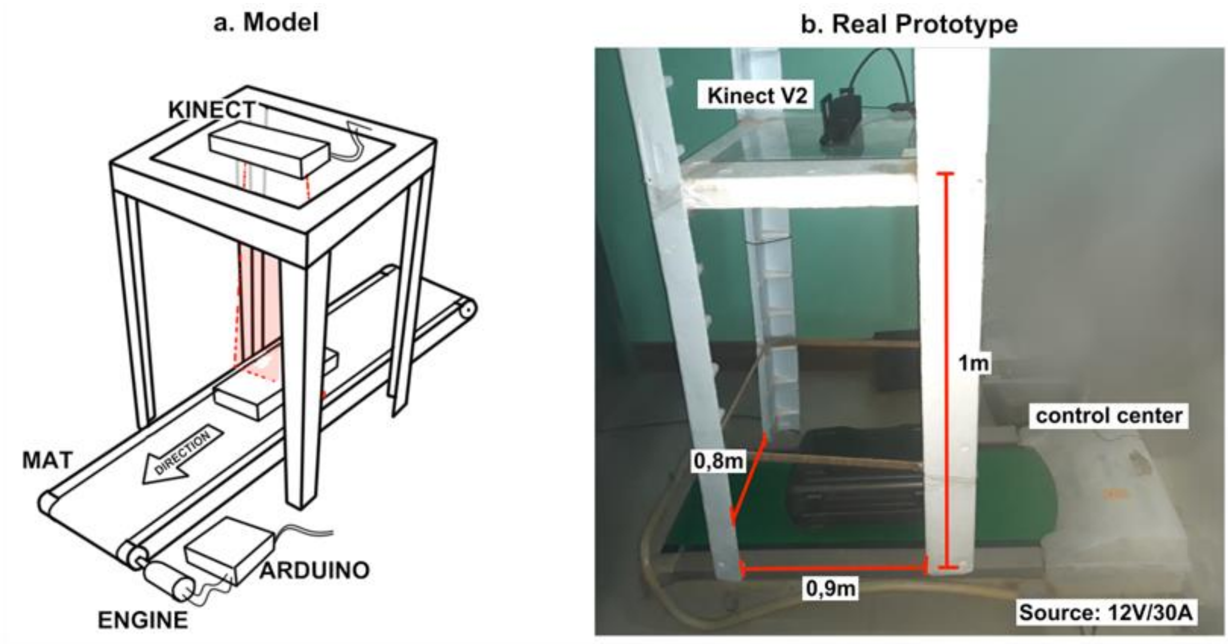


Figure 3. (a) System Model (b) System Implementation

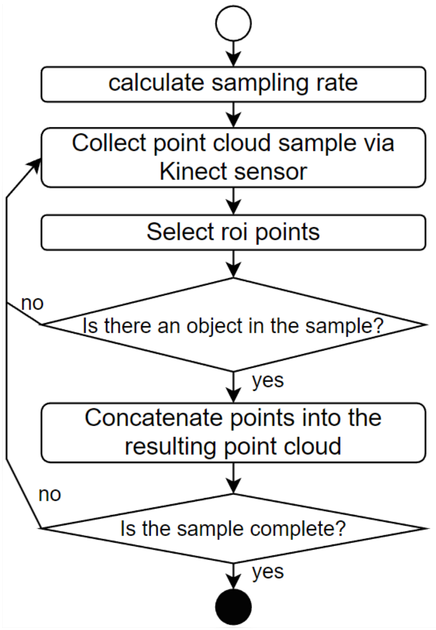


Figure 6 mobile sampling flowchart

Finally, after capturing the points we join them together and return the luggage's point cloud.

3.2. PROTOTYPE HARDWARE

Several electronic components were used to build the entire system. Precisely, we used a DC motor and an H bridge module to control the treadmill. We also employed an Arduino to enable user input. Furthermore, we reused an existing treadmill to build our structure. The structure aims to support both types of luggages: carry-on and checked luggage.

The modeling and simulation of the control center circuit were done in Proteus, a system designed to build and analyzes circuits. In the end, the physical circuit was placed in a box for protection. The developed code allows controlling the treadmill through buttons with the following options:

- Control of the direction of rotation, clockwise or counterclockwise;
- Increase treadmill speed.
- Reduce treadmill speed;
- Emergency button that stops the treadmill.

3.3. DIMENSION EXTRACTION ALGORITHM

According to the data from the systematic review, the most used algorithms for extracting

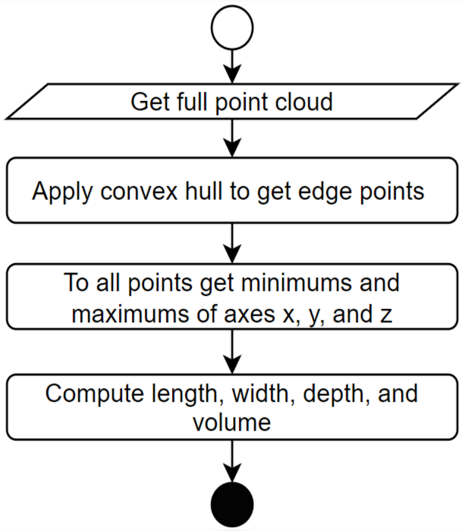


Figure 7 Algorithm used to extract luggage's dimension (AABB + Convex hull)

object dimensions are the convex hull and least squares. The least squares method approximates the shape of the point cloud to a parallelepiped, which provides a measurement that is similar to what is done in airport practice. For instance, some airport uses an empty box template to verify the luggage's conformity. Altough simple, the challenge of this methods lies in the position, quantity, and deformation of the luggage.

We implemented a simplified method that uses an axes-aligned minimal box (AABB) and the convex hull. This method returns the volume, and dimensions faithful to the format and within the measurements of a parallelepiped. We assume that the luggage has a cuboid shape. After that, we aligns the minimum and maximum axis to compute each dimension length. Figure 7 indicates the steps used by our algorithm in a flowchart.

The flowchart in Figure 7 shows that after computing the wrapping polygon we extract its boundary by analyzing the minimum and maximum points at each axis (x, y, and z). Using this data, we align them, and compute the luggage's dimension by projecting these limits in a parallelepiped. Equation 2, 3, 4, and 5 shows this computation. Here, L is the length, W is the width, and D is the depth, while h is the sensor height value which is a known constant.

$$L = (Y_{max} - Y_{min}) * 100 \qquad (2)$$

$$W = (X_{max} - X_{min}) * 100 \qquad (3)$$

$$D = (h - Z_{min}) * 100 \qquad (4)$$

$$Volume = (C * L * P) / 100^3 \qquad (5)$$

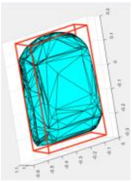
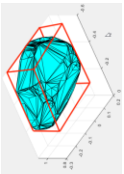
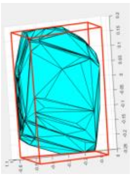
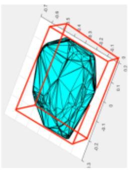
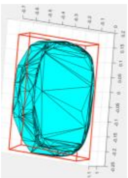
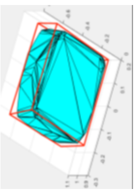
4. RESULTS AND DISCUSSIONS

To validate the prototype, we conduct tests with different luggages and bags. Table 1 shows the results obtained for the measurements, as well as the root-mean-square error (RMSE) returned for width, length, and depth. The mean absolute error (MAE) was also included to obtain insights into the dimensions individually. For each luggage, we executed 10 tests. The values presented in our approach are the average between the ones obtained in these tests.

We compare our result to the ground-truth. As such, each item is listed in terms of its width, length, and depth for the ground-truth (R), while our values are marked with (K). The N column represents the identifier code of each luggage.

From the analysis of Table 1, it is possible to see that for the L1 luggage, positioned horizontally, an RMSE of 0.30 cm was obtained. The L2 backpack, on the other hand, obtained an RMSE of 1.24 cm. Thus, it is possible to notice that there are errors with larger and smaller values. This is influenced, in addition to the sampling and position of the bag, by points slightly shifted due to reflection from the sensor when colliding with the bag's surface.

Table 1 Comparison between real-world measurements and the ones obtained by our approach

N	*Luggage*	*Convex hull and AABB*	*Real (R)*				*Our approach (K)*				*RMSE*
			Width	*Length*	*Depth*	*Volume*	*Width*	*Length*	*Depth*	*Volume*	
L1			42.50	50	17.80	0.037	42.58	50.24	17.35	0.0371	0.30
L2			42.10	55.23	18.50	0.043	42.60	56.98	19.65	0.0476	1.24
L3			42.50	50	17.80	0.037	42.96	52.04	17.21	0.0384	1.25
L4			42.10	55.23	18.50	0.043	43.64	50.04	18.29	0.040	3.13
L5			37.30	51.80	20.10	0.038	38.97	52.88	19.76	0,040	1.16
L6			41.00	58.01	22.00	0.052	44.71	59.14	21,65	0.057	2.25
	MAE						1.33	1.90	0.52	0.0027	

Another fact that corroborates the errors is the soft material of the L2 backpack. When handled, soft materials can be deformed, changing its dimensions. As we conducted multiple tests, the L2 luggage was repositioned several times, which can possibly explain the slightly different results.

The L3 and L4 tests were performed with the bags rotated diagonally. The errors returned were 1.25 cm and 3.13 cm respectively. In comparison with the results obtained in L1 and L2, there was an increase. A possible explanation is that the error is due to the projection done in AABB. Another issue is the difficulty in joining the point cloud caused by a rotated object.

Tests for L5 and L6 returned RMSE of 1.16 cm and 2.25 cm, respectively. The largest absolute error of these measurements occurred for the length values, 1.12 cm. By comparing the results of L5 and L6 with L1, it is possible to notice that the error increases according to the luggage size. A possible solution is to increase the sampling step and use extra filtering methods.

As for the time spent per measurement, the system uses an average of 0.14 s/cm. This means that, for example, given a suitcase with a length of 80 cm (the largest measure according to ANAC for dispatch luggage), the system would spend 11.8 seconds. Since the measurement time directly influences the check-in, it is interesting to make improvements to the system to reduce this time, such as reducing the time required to handle each sample and increase the treadmill speed.

Analyzing all tests, the mean absolute error was 1.33 cm for width, 1.90 cm for length and 0.52 cm for depth, totaling an average absolute error of 1.25 cm.

5. CONCLUSIONS

The present work proposed a model for detecting airport luggage dimensions using a depth sensor. Therefore, a systematic review was carried out, which found that the leading technologies used in the market are laser scanners, binocular vision, and depth sensors. The limitations found were the position, quantity, and shape of the luggage.

So, a prototype was built. The test results shown in session 4 indicate that the system was able to obtain the point cloud and calculate the dimensions of the luggage with an overall MAE of 1.25 cm. The average time expend for measurement is 0.14 s/cm. This indicates that the system needs improvements in time consuming. These results demonstrate that there is potential for using this low-cost alternative model in managing airline boarding operations and investments. By reducing measurement errors, it is possible to optimize the use of space on the aircraft and alleviate passenger frustrations.

In next stages of this research, is intended to deepen the tests regarding other positions and the amount of luggage by measurements, as well as explore the impact of handles, wheels, and labels on the results. It is also intended to explore obtaining the weight of the luggage.

References

Alsyouf, I., Kumar, U., Al-Ashi, L., & Al-Hammadi, M. (2018). Improving luggage flow in the luggage handling system at a UAE-based airline using lean Six Sigma tools. Quality Engineering, 30(3), 432–452.

ANAC. (2022). Passageiros. Agência Nacional de Aviação Civil (ANAC). Available at: https://www.gov.br/anac/pt-br/assuntos/passageiros. Accessed on 19/10/2022.

Anderson, K. (2019). Singapore Changi Airport caught using Microsoft Kinect sensors. Onmsft.com. Available at: https://www.onmsft.com/news/singapore-changi-airport-caught-using-microsoft-kinect-sensors. Accessed on 19/10/2022.

Azul. (2022). Bagagem Despachada. Azul. https://www.voeazul.com.br/para-sua-viagem/informacoes-para-viajar/bagagem-despachada

Bhowmik, D., & Appiah, K. (2018). Embedded Vision Systems: A Review of the Literature. Applied Reconfigurable Computing. Architectures, Tools, and Applications, 204–216.

Bourque, P., & Fairley, R. (2015). Guide to the Software Engineering Body of Knowledge SWEBOK ® A Project of the IEEE Computer Society.

Chan, T., Lichti, D., Jahraus, A., Esfandiari, H., Lahamy, H., Steward, J., & Glanzer, M. (2018). An Egg Volume Measurement System Based on the Microsoft Kinect. Sensors, 18(8), 2454.

Colby, M. (2019). Self-service bag drops and the challenges of speeding up airport luggage check-in. Available at: https://www.stantec.com/en/ideas/self-service-bag-drops-and-the-challenges-of-speeding-up-airport-luggage-check-in. Accessed on: 19/10/2022.

Ding, S., Nie, X., Qiao, H., & Zhang, B. (2018). A Fast Algorithm of Convex Hull Vertices Selection for Online Classification. IEEE Transactions on Neural Networks and Learning Systems, 29(4), 792–806.

Gao, Q. J., & Yang, L. (2013). Luggage Specification Detection Based on the Binocular Vision. Applied Mechanics and Materials, 278-280, 861–865.

Gao, Q., & Liang, P. (2021). Airline Luggage Appearance Transportability Detection Based on A Novel Dataset and Sequential Hierarchical Sampling CNN Model. IEEE Access, 9, 41833–41843.

Gao, Q., Yin, D., Luo, Q., & Liu, J. (2018). Minimum elastic bounding box algorithm for dimension detection of 3D objects: a case of airline luggage measurement. IET Image Processing, 12(8), 1313–1321.

Göçmen, E. (2021). Smart Airport: Evaluation of Performance Standards and Technologies for a Smart Logistics Zone. Transportation Research Record: Journal of the Transportation Research Board, 2675(7), 480–490.

Gol. (2022). Bagagem | GOL Linhas Aéreas. Available at: https://www.voegol.com.br/informacoes/bagagem. Accessed on: 19/10/2022.

Kosacka-Olejnik, M., & Pitakaso, R. (2019). Industry 4.0: state of the art and research implications. Logforum, 15(4), 478–485.

Kuan, Y. W., Ee, N. O., & Wei, L. S. (2019). Comparative Study of Intel R200, Kinect v2, and Primesense RGB-D Sensors Performance Outdoors. IEEE Sensors Journal, 19(19), 8741–8750.

Limwattanapibool, O., & Arch-int, S. (2017). Determination of the appropriate parameters for K-means clustering using selection of region clusters based on density DBSCAN (SRCD-DBSCAN). Expert Systems, 34(3)

Moher, D., Liberati, A., Tetzlaff, J., & Altman, D. G. (2009). Preferred Reporting Items for Systematic Reviews and meta-analyses: the PRISMA Statement. BMJ, 339.

Negri, N., & Borille, G. (2017). Avaliação da influência de novas tecnologias em terminais de passageiros aeroportuários sob a ótica dos passageiros. Anais do XXXI Congresso em Pesquisa e Ensino em Transportes, ANPET.

Qingji, G., Chuanbo, P., & Qijun, L. (2018). Method on 3D Reconstruction of Airline Luggage Based on Active Laser Projection of Improved Encoding. 2018 IEEE CSAA Guidance, Navigation and Control Conference (CGNCC), 39.

Ren, X., Zhou, X., & Xu, X. (2020). A new model of luggage storage time while boarding an airplane: An experimental test. Journal of Air Transport Management, 84, 101761.

Ronzani, G. M., & Correia, A. R. (2015). Impact of demand and airport characteristics on luggage claim. Proceedings of the Institution of Civil Engineers - Transport, 168(2), 150–160.

Ruchay, A. N., Dorofeev, K. A., & Kolpakov, V. I. (2018). Fusion of information from multiple Kinect sensors for 3D object reconstruction. Computer Optics, 42(5), 898–903.

Statista. (2021, October 27). Airline industry - passenger traffic worldwide 2019 | Statistic. Statista. Available at: https://www.statista.com/statistics/564717/airline-industry-passenger-traffic-globally/. Accessed on: 19/10/2022.

Zennaro, S., Munaro, M., Milani, S., Zanuttigh, P., Bernardi, A., Ghidoni, S., & Menegatti, E. (2015). Performance evaluation of the 1st and 2nd generation Kinect for multimedia applications. 2015 IEEE International Conference on Multimedia and Expo (ICME), 1–6.