Towards a Robust Solution for the Supermarket Shelf Audit Problem

Emmanuel F. Morán[a], Boris X. Vintimilla[b] and Miguel A. Realpe[c]
ESPOL Polytechnic University, Escuela Superior Politecnica del Litoral, ESPOL, CIDIS,
Campus Gustavo Galindo Km. 30.5 Vía Perimetral, P.O. Box 09-01-5863, Guayaquil, Ecuador

Keywords: Retail, Supermarket, Shelves Auditing, Deep Learning, Supermarket Dataset.

Abstract: Retail supermarket is an industrial sector with repetitive tasks performed using visual analysis by the store's operators. Tasks such as checking the status of the shelves can contain multiple sequential sub-tasks, each of which needs to be performed correctly. In recent years, there has been some intents to create a solution for the tasks mentioned without been complete solution for retails. In this article, a first realistic approach is proposed to solve the supermarket shelf audit problem. For this, a workflow is presented to deliver compliance level with respect to the expected store's planogram.

1 INTRODUCTION

In recent years, AI has begun to be implemented in a wide number of fields, including medicine (Chua et al., 2021), agriculture (Zhang et al., 2020), finance (Patil et al., 2022), among others. We address a topic that to date and to the best our knowledge, has not been fully analyzed in the public literature: Retail Supermarket Shelves Auditing.

Shelves Auditing can be defined as the process of comparing the current state of shelves against the expected state according to a planogram (a visual model for distributing supermarket products on their respective shelves). To make this comparison, human operators must carry out visual checks following the protocol of the store in order to validate the status of the shelves. As result of these visual processes, a percentage of compliance between the actual shelves and the planogram is calculated. Finally, by averaging these values of all the shelves in the sales area, a store compliance percentage can be obtained.

In this article, a new proposal to solve the supermarket shelf audit problem by defining an acquisition method for a new dataset and a workflow to process it will be discussed.

[a] https://orcid.org/0000-0001-6915-7370
[b] https://orcid.org/0000-0001-8904-0209
[c] https://orcid.org/0000-0001-8711-5596

2 SUPERMARKET ISSUES

Many problems in retail supermarkets have been identified and reported in the literature (Pettigrew et al., 2005) (Li and Wang, 1970) (Jedlickova, 2016). This section will focus on the issues that involve product placement in the shelves, since these problems are related to the fact that product positioning is done by humans, inevitably leading to some errors.

2.1 Outdated Price Tags

This problem refers to price tags that are outdated (printed or digital) or are in poor condition (torn, stained, poorly printed, etc.). Price tags are one of the most important objects on the shelves, since they allow to know what product is being displayed, its description, weight or size and price. If price tags are out of date, it is impossible for customers to correctly identify the product or to know the real price of the product. An outdated price on the price tag can have two negative repercussions: If the actual price is lower than what is displayed (for example, a product that should have a special offer), it is probably not attractive to customers, and therefore sales may be lost. On the other hand, if the real price is higher than the one shown, the customer may feel cheated when checking out. This last one could have legal repercussion as it can be consider a consumer scam. An option for this to not scale is to sell the product at the seen price (losing money).

A conventional solution could be digital/electronic price tags (Cochoy and Soutjis,

Morán, E., Vintimilla, B. and Realpe, M.
Towards a Robust Solution for the Supermarket Shelf Audit Problem.
DOI: 10.5220/0011747000003417
In *Proceedings of the 18th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2023) - Volume 4: VISAPP*, pages 912-919
ISBN: 978-989-758-634-7; ISSN: 2184-4321

2020). These are price tags tag can massively change their content (price, description, code, etc), but sadly implementing this is expensive, and most of the retails could not afford. Someone could also think to use RFID technology to easily catch information from a price tag, but this technology is not conventional for this particular use since it is normally use for identifying product in the supply chain (Tao et al., 2022; Škiljo et al., 2020) and ease inventory in categories with high margin of gain like clothes and bottles with alcoholic beverages such as whiskey(Cilloni et al., 2019; Khalil et al., 2020).

2.2 Inactive Assortment

Inactive Assortment is the denomination given to products that should be displayed in the sales area (space where shelves are, and products are displayed), but for some reason are not. Some of the reasons for this to happen are: out of stock, stock on the way but not yet in store, or store decision. Inactive assortment could drive shoppers away as they can't find some products.

2.3 Product Breaks

Denomination for a partially displayed product on the shelves. This is considered bad for the presentation because the shelf appears to be empty. This is different from inactive assortment because product is displayed in the shelf tray (shelf area for the product) but is partially empty. This could happen for many reasons, such as out of stock, replenishment not yet completed, or simply the product is in stock but not yet displayed on the shelves due to operator error. The latter is the most common mistake and can be fixed by manually checking all the products on the shelves.

2.4 Products in Poor State

A product state is how it is displayed in the shelf and the envelope of it. If a product is in poor state it could drive to a customer to not perform the purchase and even do not return to the store ever again. This could be fault of operators who have not followed the store protocols or perhaps some children playing around in the store.

3 THE PROCESS OF SHELVES AUDITING

In a supermarket, shelves auditing can be defined as the process of comparing the current state of the shelves with the expected state according to its planogram. This comparison is necessary for many aspects, among them, validating contractual compliance with suppliers who paid extra for strategic positioning of their products on the shelves, or to be able to carry out market analysis, i.e. how certain products behave if they are placed closer to or further away from others.

Such comparisons are usually made by supermarket operators, and like any repetitive process performed by a human being, it is prone to errors and time consuming due to its visual nature.

The following states must be validated during the audit of the supermarket shelves:

- Presence of all products on each shelf according to the planogram.
- Presence of product price tags in the correct position.
- Concordance between the product and its price tag according to the defined protocol.
- Product price updated in the price tag.
- Presence of product breaks.
- Product status (correct number of shelf tray fronts, well-organized products, or others that the supermarket may need).

All validations mentioned are performed during store opening hours, but these are more extensive before or after hours of operation due to the absence of customers. The operator will first check the shelves in all aisles and try to solve problems like wrongly positioned price tags or products. For other issues, such as product breaks, the operator first will go to the warehouse to pick up missing products and then will return to fill the shelf. This process is tedious and repetitive enough to lead operators to make mistakes. For this reason, there is a need to automate the step of reviewing the shelves and looking for existing problems in them, thereby the operators have this information directly and can attend to them with quick and precise solutions.

4 DISCUSSION: A WAY FORWARD

The remarkable work presented by (Goldman et al., 2019a) with their dataset named SKU110k (Goldman et al., 2019b) has fueled initiatives for shelves auditing solutions such as (Chen et al., 2022) with their dataset named UniTail, that also includes other processes like text detection and recognition, but do not

Figure 1: Examples of price tags from the SKU110K and UniDet datasets, from left to right respectively.

provide all the means to solve the problems mentioned above in Section 2. For this, it might be better to create a dataset capable of addressing all the requirements in shelf auditing.

As outlined earlier, the price tag is an important object on the shelves, because it shows the price and description of the product and is used for limiting a product's front space, helping to understand if the product was positioned correctly on the shelves. With this in mind, datasets like SKU110k and UniDet cannot be used to validate the mentioned states since not all data within the price tag can be collected (reading it, as seen in Figure 1, which means it is not possible to solve the problem of outdated prices because it is impossible to match the object itself with the databases where retailers save and update product prices for transactions at checkout.

Furthermore, when using the product package text for matching, as done in (Chen et al., 2022), there will be a considerable number of products that could not be correctly identified. This is indeed a weakness of this work. On the other hand, when using the product price tag, there will always be a full description of the product, or at least a code to identify it.

Lastly, not having the price tag localization will prevent the pipeline from reporting some shelf audit issues, such as wrongly positioned products.

Current public datasets are the problem because they were created with little or no knowledge of what is needed to solve the retail problem: Shelf Audit. Trying to overcome the whole problem can end up with pipelines of craft solutions, not generalizable solutions.

4.1 Proposed Dataset

As far as is known, all the approaches that have been proposed try to solve the shelf audit problem using only RGB images. However, as mentioned above this is a limitation. For this reason, a new dataset to meet all the needs of this problem is proposed in this section.

For the dataset, the following aspects should be considered:

1. **RGB-UHD Images.** RGB is the standard Red-Green-Blue convention for images, while UHD means Ultra High Definition images. Figure 2 (left) shows an example for this type of images, which are captured using a high definition camera (preferably a resolution of 4k or higher). In this dataset, the information to be tagged is: the product and the price tags.

Figure 2: Example of proposed images; left: RGB-UHD image, and right: depth image (this image have been binarized).

2. **Depth Images.** these images are captured by a 3D camera and allow to obtain the measured depth from the camera lens to the objects (normally a resolution of 640x480 pixels), see Figure 2 (right). No need for annotations.
3. **Positional Information.** defines the position or location information relative to the acquisition site. This information will help understanding where the image is taken on the store map and can be delivered in [x,y,z] format, which means the (x,y) position on the map at (z) meters above the floor.
4. **Planograms.** a structure that contains the expected position of the product on the shelves in order to compare it with the current position.
5. **Master Database.** a database to consult information about the products such as: prices, descriptions, codes, among others. This is common information that all retailers should have.

4.2 Proposed Acquisition System for the Dataset

For acquiring the proposed dataset, an automated acquisition system is presented (which from now and on will be referred as "robot"). This is crucial for reducing the time consumed by the operators in auditing the shelves. Using operators to acquire the information will only add more tasks to them, increasing time instead of reducing it.

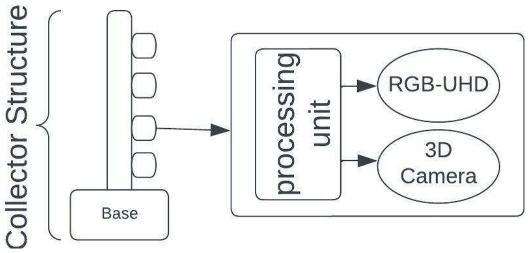

Figure 3: Collector Structural Design.

Figure 3 shows the proposed robot. It is not different from others already deployed in retail. It is based on a mobile base (2 or 4 wheels with suspension system, battery, and voltage regulation), with a structure on it, similar to a tower, where the collectors are vertically and equidistantly placed. The collectors consist of low computational resources processing units (Raspberry Pi 4, Jetson Nano, Intel Nuc, among others), a RGB-UHD camera and a 3D Camera.

This robot needs to move autonomously, so it is recommended to use state-of-the-art robotic software for moving it, like ROS(Stanford Artificial Intelligence Laboratory et al.,), ROS2(Macenski et al., 2022). This software is able to estimate the positional information of the robot with respect to the store map at all times.

An important part of the robot is the way it captures the information. It goes through the hall of the store taken images with the cameras of collectors. This route is performed by doing *steps*. A set of images (rgb-uhd and depth images per collector) is taken every step. Every step is limited by the fields of views (FOVs) of the cameras. This is done because the main purpose is to acquire information, so redundancy is introduced during the acquisition. For *horizontal redundancy* in the images, the distance of each step is shorter than the FOV of the cameras, and for *vertical redundancy*, the collectors are equidistantly positioned in the tower of the robot. Have redundant information provides the certainty of not losing information at all.

4.3 Proposed Pipeline

All processes, including the inputs and outputs, are explained below. Figure 4 shows the pipeline of the proposed solution for the shelf auditing problem. Blue and green blocks are inputs (data acquired as mentioned before) and outputs (reports) respectively. Gray blocks are processes that may involve the use of artificial intelligence algorithms such as object detection, object recognition, clustering, text recognition, among others. Yellow blocks are also processes but guided to validations or estimations, that is, they use the manipulated and filtered data for creating the report outputs.

It most be mentioned that the proposed pipeline does not try to add new hardware or processes to the stores, like in the case of implementing digital/electronic price tags or RFIDs to the price tags or products, since this will create new expenses to the retail.

In Figure 5 is shown how the RGB and depth images are processed during the firsts blocks in the pipeline. Yellow section refers to product detection and recognition; blue section refers to price tags detection, items detection and recognition; while the green section refers to gap detection.

Product Detection. This process requires RGB-UHD images as input and produces RBOXs that represent the products as outputs. Each RBOX is defined by a 7-value list containing the information of the detection (x axis, y axis, height, width, rotation angle, confidence and class). A pre-trained object detection algorithm could be used to carry out this process. To continue with other processes, crops of the products should be done, and will be referred as *uhd-product images*.

Product Recognition. This process requires uhd-product images as input and produces a text referring to the product class. The output can be represented by a description or a code, but it is recommend to use codes instead of description, as the dataset will be lighter (meaning size, as the code is normally a shorter string), additionally, the probability of changing products' descriptions is higher than the one of changing codes. A pre-trained multi-classification algorithm would be needed to carry out this process.

Price Tag Detection. This process requires RGB-UHD images as input and produce RBOXs that represent the price tags as outputs. A pre-trained object detection algorithm for the task of detecting price tags could be used to carry out this process A custom dataset must be created for this process, since to the best of our knowledge there is no public dataset created for detecting price tags. To continue with other processes, crops of the price tags should be done, and will referred as *uhd-price-tag images*.

Note that, there is the option of creating a single detector for price tags and products, but this pipeline shows them separated just to make clear each process of the pipeline.

Items Detection. This process requires as input uhd-price-tag images and produce RBOXs that represent the *items* of the price tags. Items is the denomination in this article to the important texts in the price tags such as, but not limited to, codes, prices and descriptions. A pre-trained object detection algorithm for the

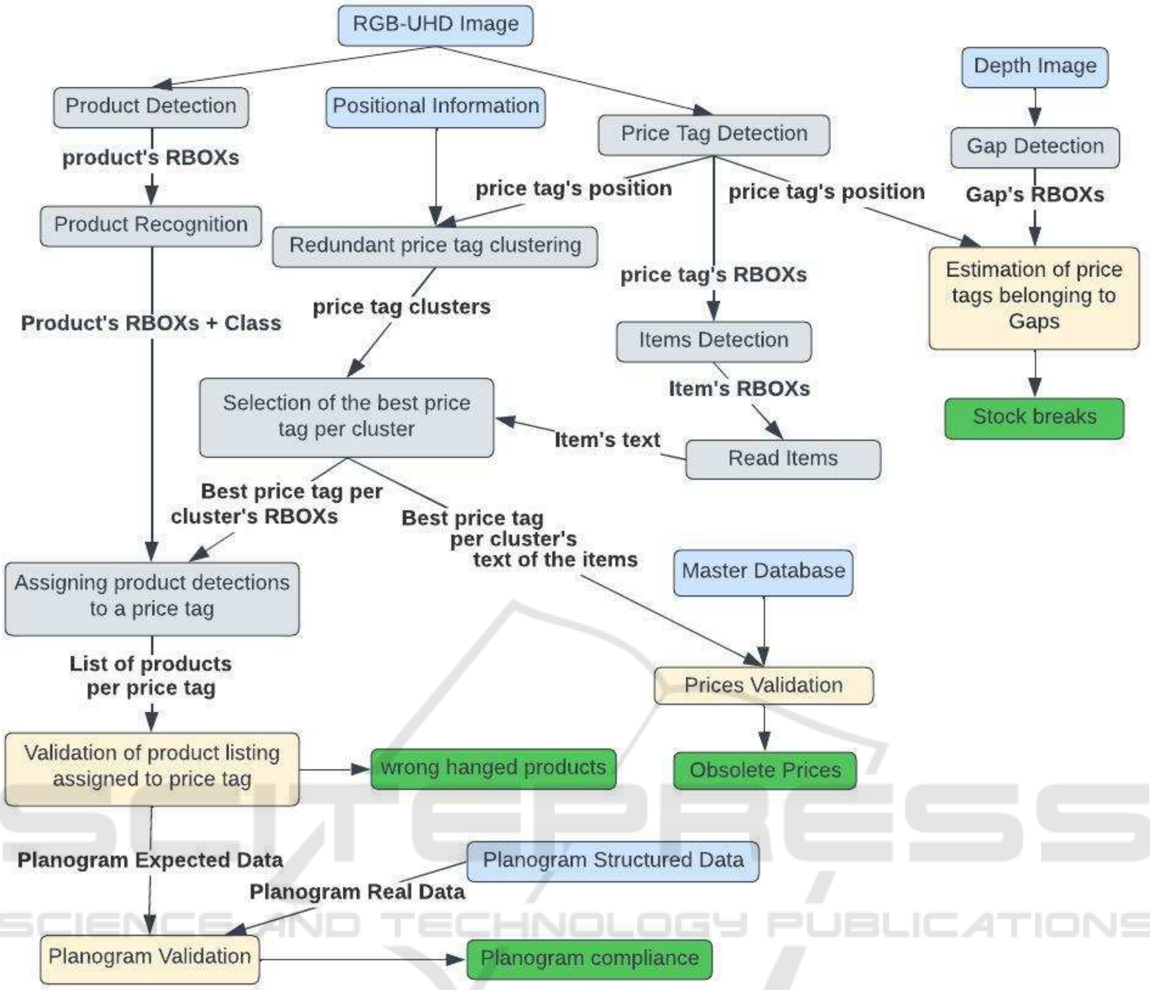

Figure 4: Proposed pipeline to solve the shelves auditing problem.

task of detecting items could be used to carry this process and deliver RBOXs of the detection as outputs, such as text detectors. To continue with other processes, crops of the items should be done. We will refer to these as *uhd-items images*.

Read Items. This process requires as input the uhd-items images and produce texts as outputs. The detected items of each price tag must be read in order to know what information is shown and subsequently contrast that information against the *Master Database*. The items mentioned are texts, thus the way to read those items is by OCR, for other items like the barcode a barcode-reader should be implemented (Code 39, Code 128, GS1-128, EAN-13, among others). To continue with other processes, the texts generated will be referred as *items text*.

Gap Detection. This process requires the input of depth images and produce RBOXs that represent the gaps in the shelves. Each pixel of the depth image contains the depth measured from the 3D camera lens. The RBOXs are relative to the depth image, and should be projected to the rgb-uhd images. For accomplish this, a conversion using a translation matrix should be done.

Redundant Price Tag Clustering. To be able to select the best of all the price tags (next process) it is important first to correctly group all the redundant instances of the same object, and then proceed to choose the best one that represents the real instance of the object. For this process, spatial information related to the collection location is required, that is, the information of the position where each image is collected is also saved with respect to the reference system of the store. With the help of rotational and translations transformations, it is possible to obtain the position of each object's centroid (x,y) with respect to the reference system. Finally, using a clustering algorithm that will have as input the spatial positions of the price tag, the clusters can be obtained. Initial cluster could fail given the dense environment in a shelf, so it is recom-

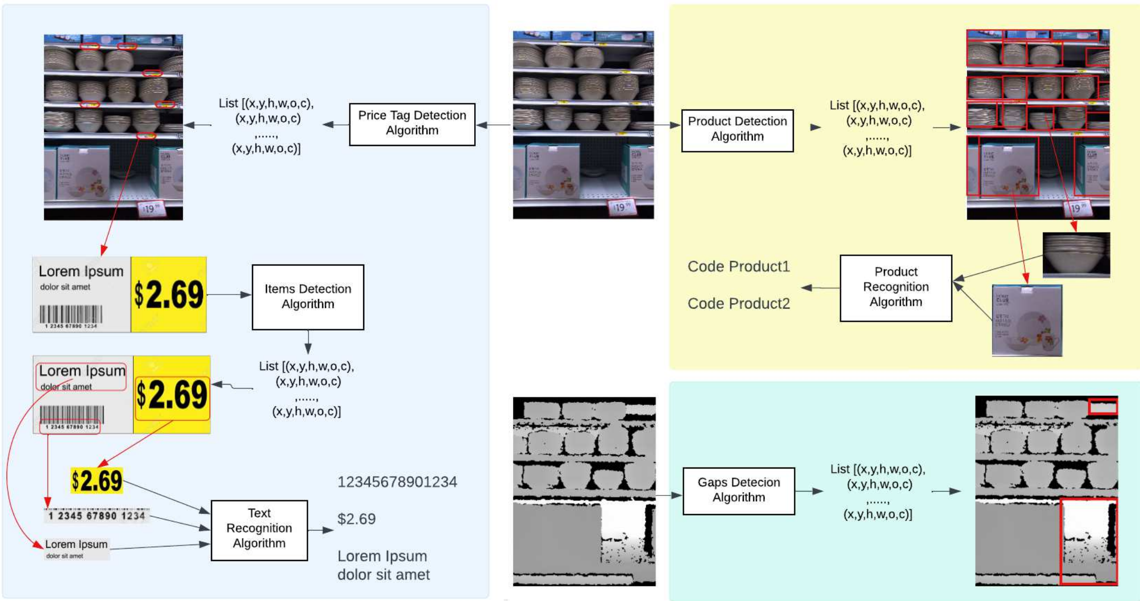

Figure 5: Product Detection Process of the pipeline.

mended to use a subsequent algorithm to validate the items inside the cluster using the items text of each price tag.

Selection of the Best Price Tag per Cluster. Being redundant means that multiple captures were obtained for the same price tag, and therefore it is necessary to choose the best of all to be the representative of the group. The selection of this can be varied, but it is suggested to do it according to the characteristics of objectness of the price tag and the confidence of readings of the items belonging to each price tag using a weighted average as the metric for ordering. The output of this process is a list of the price tags.

Prices Validation. The texts of the best price tags of each generated cluster are required as input and to create a list of products. Then, a comparison between this list and the data on the master database is done. The output of this validation process is a new list of products that have obsolete prices (different from the price in the database). This report is known as ”obsolete prices”.

Assigning Product Detections to a Price Tag. This process is quite significant. It serves to spatially group the products with a specific price tag. Taking into account the common protocol for the location of each price tag by product indicated in section 2, an algorithm can assign each detected product instance to a particular price tag. A case in Figure 6 can be seen, price tag 6 assigned with the products with purple colour. The way to perform this is by first filtering all the possible price tags by using the product centroid and the price tags centroid, only the price tags that

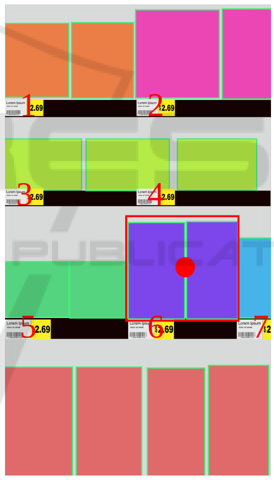

Figure 6: Positioning of the price tags in a shelf.

are at the left and down the centroid of the product continue. Then, the minimum Euclidean distance between each price tag and the product is selected as the price tag for the product, and the product is assigned to the price tag. This is simple, yet effective to assign the product detections to labels. It should be noted that in the case of the hanged products in the shelving hooks, the price tags are on top of the product. So the logic in this algorithm can be modified. For products in the borders that do not have an assigned price tag, they will be discarded on the premise that being in the border of the images, it is likely that a previous, subsequent, spatially higher or lower collection iteration contains the complete vision of the product together to the price tag in order to be assigned correctly.

Estimation of Price Tags Belonging to Gaps. Similar to assigning product detection to a rule, in this case the previously detected gaps will be assigned to a particular price tag. In the case of posts, this process is also valid, however the reader is reminded that for spaces with posts, it is more likely that there are gaps that are not actually used for products. This process ends by delivering a report of all the gaps found, the location (hall) and an estimation of the product that can be according to the price tag to which it was assigned. This report is known as ”stock breaks”.

Validation of Product Listing Assigned to Price Tag. This process requires as input the list of products per price tag instance (best of each cluster). Each Product (RBOX) must match the price tag (product recognition output vs item recognition output). In case of having products that do not agree with the others, they will be used to carry out the report of Mis-Positioned Products. This report will deliver a list of the products and their location (spatial) in the aisle in order to quickly identify and correct them. Its output can also be modified to a structure called a planogram structure. This output is known as the ”real planogram structure”.

Planogram Validation. This process requires two planogram data structures as input. The real planogram structure (EPR) and the expected planogram structure (EPE). The EPR represents what is real on the premises, while the EPE represents what should be implemented. Ideally, EPR equals EPE, however stores are likely to make changes mistakenly or intentionally. Therefore, the comparison of these two structures will deliver a percentage value of planogram compliance. The comparison can be made according to the purpose to be measured, be it at a granular level of product, categories or others.

5 FUTURE WORK

At the moment, the implementation of the acquisition system and datasets have been accomplish gratefully thanks to a retail company which is interested in replicating this project in many stores. The first attempts to create the obsolete prices report implementing the light blue side of Figure 5 has given satisfactory results, but we will continue working on this for better results. Figure 7 shows results of the implementations.

Additionally, we will be working in parallel in the hot topic of product detection and recognition for closing the gaps between the product and the price tag’s information and location in the shelf.

Figure 7: Results of implementation.

6 CONCLUSIONS

Retail is an important sector where AI can help to reduce manual and repetitive tasks. To date, AI is mature enough to be implemented in real world projects. We proposed a challenging but reachable pipeline to solve one of the biggest problems in retail: shelves auditing, to help the retail’s operators in their daily tasks for maintaining the store at it best for customers. Having defined a pipeline, future work is to create it and test it. This might sound easy, but it is far from being the case. There are many complications like creating the acquisition system or creating agreements with retailers to gather data to elaborate, annotate and publish datasets.

ACKNOWLEDGEMENTS

This work has been partially supported by the ESPOL-CIDIS-11-2022 project and Tiendas Industriales Asociadas Sociedad Anonima (TIA S.A.). The authors would like to acknowledge TIA S.A., a leading grocery retailer in Ecuador, for providing access to an incredible environment for research and testing.

REFERENCES

Chen, F., Zhang, H., Li, Z., Dou, J., Mo, S., Chen, H., Zhang, Y., Ahmed, U., Zhu, C., and Savvides, M. (2022). Unitail: Detecting, reading, and matching in retail scene.

Chua, I. S., Gaziel-Yablowitz, M., Korach, Z. T., Kehl, K. L., Levitan, N. A., Arriaga, Y. E., Jackson, G. P., Bates, D. W., and Hassett, M. (2021). Artificial intelligence in oncology: Path to implementation. *Cancer Medicine*, 10(12):4138–4149.

Cilloni, G., Leporati, R., Rizzi, A., and Romagnoli, G. (2019). State of the art of item-level rfid deployments in fashion and apparel retail. *International Journal of RF Technologies*, 10:65–88. 3-4.

Cochoy, F. and Soutjis, B. (2020). Back to the future of digital price display: Analyzing patents and other archives to understand contemporary market innovations. *Social Studies of Science*, 50(1):3–29. PMID: 31630628.

Goldman, E., Herzig, R., Eisenschtat, A., Goldberger, J., and Hassner, T. (2019a). Precise detection in densely packed scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Goldman, E., Herzig, R., Eisenschtat, A., Ratzon, O., Levi, I., Goldberger, J., and Hassner, T. (2019b).

Jedlickova, B. (2016). Vertical issues arising from conduct between large supermarkets and small suppliers in the grocery market: Law and industry codes of conduct.

Khalil, G., Doss, R., and Chowdhury, M. (2020). A novel rfid-based anti-counterfeiting scheme for retail environments. *IEEE Access*, 8:47952–47962.

Li, B. and Wang, D. (1970). Configuration issues of cashier staff in supermarket based on queuing theory.

Macenski, S., Foote, T., Gerkey, B., Lalancette, C., and Woodall, W. (2022). Robot operating system 2: Design, architecture, and uses in the wild. *Science Robotics*, 7(66):eabm6074.

Patil, D., Patil, S., Patil, S., and Arora, S. (2022). Financial forecasting of stock market using sentiment analysis and data analytics. In Nagar, A. K., Jat, D. S., Marín-Raventós, G., and Mishra, D. K., editors, *Intelligent Sustainable Systems*, pages 423–430, Singapore. Springer Nature Singapore.

Pettigrew, S., Mizerski, K., and Donovan, R. (2005). The three "big issues" for older supermarket shoppers.

Stanford Artificial Intelligence Laboratory et al. Robotic operating system.

Tao, F., Wang, L., Fan, T., and Yu, H. (2022). Rfid adoption strategy in a retailer-dominant supply chain with competing suppliers. *European Journal of Operational Research*, 302(1):117–129.

Zhang, Q., Liu, Y., Gong, C., Chen, Y., and Yu, H. (2020). Applications of deep learning for dense scenes analysis in agriculture: A review. *Sensors*, 20(5).

Škiljo, M., Šolić, P., Blažević, Z., and Perković, T. (2020). Analysis of passive rfid applicability in a retail store: What can we expect? *Sensors*, 20(7).