Multi-Class Product Counting And Recognition for Automated Retail Checkout: A survey paper of The 6th AI City Challenge Track4

Arpita Vats
Santa Clara University
Santa Clara, CA, USA
avats@scu.edu

Abstract

Track 4 of of the 6th AI City Challenge specifically focuses on implementing accurate and automatic check-out systems in retail stores. The challenge includes identifying and counting products as they move along a retail checkout conveyor belt, despite obstacles such as occlusion, movement, and similarity between items. I was on the evaluation team for this track where I evaluated the methods of the top-performing teams on hidden Testset B along with my professor, David C. Anastasiu, who is on the organizing team of the challenge. Teams were provided with a combination of real-world and synthetic data for training and were evaluated on their ability to accurately recognize products in both closed and open-world scenarios, as well as the efficiency of their program. The team with the highest combined score for effectiveness and efficiency was declared the winner. The goal of this track is to accurately identify and count products as they move along a retail checkout lane, even if the items are similar or obscured by hands. Distinguishing this track from others, only synthetic data was provided for training models. The synthetic data included a variety of environmental conditions to train models on, while real-world validation and test data will be used to evaluate the performance of models in realistic scenarios.

1. Introduction

Self-checkout systems are becoming increasingly popular in retail environments, as they allow customers to quickly and easily complete their transactions without the need for assistance from a store employee. In recent years, the use of deep learning techniques in self-checkout systems has become more prevalent, as they can provide higher levels of accuracy and faster processing times compared to traditional methods. One example of a deep learning-based self-checkout system is the use of object recognition algorithms to automatically identify and classify items as they are placed on the checkout scanner. These algorithms use convolutional neural networks (CNNs) to analyze images of the items and accurately identify them based on their visual characteristics. This can significantly improve the speed and accuracy of the self-checkout process, as customers no longer need to manually scan each item or input its identification code. Another application of deep learning in self-checkout systems is the use of natural language processing (NLP) algorithms to enable customers to input information using voice commands. These algorithms can understand and respond to a wide range of spoken inputs, allowing customers to easily specify the items they are purchasing, apply for discounts or promotions, and complete other tasks without the need for manual input. In addition to improving the customer experience, deep learning-based self-checkout systems can also provide significant benefits for retailers. For example, they can reduce the need for store employees to be stationed at checkout stations, allowing them to focus on other tasks such as customer assistance and inventory management. They can also provide more accurate and detailed data on customer purchases, allowing retailers to better understand consumer behavior and make more informed business decisions. Overall, the use of deep learning in retail self-checkout systems has the potential to greatly enhance the customer experience and provide significant benefits for retailers. As these technologies continue to advance and become more widely adopted, they are likely to become an increasingly important part of the retail landscape. Implementing automatic product recognition in grocery stores through images has a significant impact on the retail industry. Firstly, it will benefit the planogram compliance of products on the shelf. For instance, product detection can identify which items are missing from the shelf to remind the store staff to replenish the products immediately. It is observed that when an optimized planogram is 100% matched, sales will be increased by 7.8% and profit by 8.1% [17]. Furthermore, image-based commodity identification can be applied to automatic self-checkout systems to optimize the user experience of check-

out operations. Global self-checkout (SCO) shipments have steadily increased between 2014 and 2019. Growing numbers of SCOs have been installed to reduce retailers' costs and enhance customer experience [11]. The research [12] demonstrates that customers waiting time for checkout operations has a negative influence on their shopping satisfaction, which is to say that applying a computer-vision-based product recognition in SCOs benefits both retailers and customers. Thirdly, product recognition technology can assist people who are visually impaired to shop independently, which is conducive to their social connectivity [8]. Traditional shopping methods usually require assistance from a sighted person because it can be difficult for a person who is visually impaired to identify products by their visual features (e.g., price, brand, and due date), making purchase decisions difficult [11].

2. Background

The AI city challenge [13] specifically focuses on problems in two domains where there is tremendous unlocked potential at the intersection of computer vision and artificial intelligence: Intelligent Traffic Systems (ITS), and brick and mortar retail businesses. The four challenge tracks of the 2022 AI City Challenge received participation requests from 254 teams across 27 countries. Track 4 was another new track in 2022 aiming to achieve retail store automated checkout using only a single view camera. We released two leader boards for submissions based on different methods, including a public leaderboard for the contest, where no use of external data is allowed, and a general leader board for all submitted results. The top performance of participating teams established strong baselines and even outperformed the state-of-the-art in the proposed challenge tracks.

3. Datasets

The Automated Retail Checkout (ARC) dataset includes two parts: synthetic data for model training and real data for model validation and testing. The synthetic data for Track 4 is created using a pipeline from a specific source. This pipeline involved collecting 116 scans of real-world retail objects obtained from supermarkets in 3D models. The objects range in class from daily necessities, food, toys, furniture, household items, etc. A total of 116,500 synthetic images were generated from these 116 3D models. The images were filmed in a scenario as depicted in the provided figure. Random attributes including random object placement, camera pose, lighting, and backgrounds were adopted to increase the dataset diversity. Background images were chosen from a specific image dataset, which has diverse scenes suitable for serving as natural image backgrounds. In the test scenario, the camera is mounted above the checkout counter and facing straight down, while a customer is enacting a checkout action by "scanning" objects in front of the counter in a natural manner. Several different customers participated, and each of them scanned slightly differently. There is a shopping tray placed under the camera to indicate where the AI model should focus. In summary, approximately 22 minutes of video were obtained, and the videos were further split into test A and test B sets. The former amounts to 20% of recorded test videos that were used for model validation and inference code development. The latter accounts for 80% of the videos, which were reserved for testing and determining the ranking of participant teams.

Figure 1. The Automated Retail Checkout (ARC) dataset includes two parts: synthetic data for model training and real-world data for model validation and testing. [14]

4. Proposed Methods

The proposed methods are one of the top-performing models for the AI City Challenge 2022, Track 4. The main objective of this challenge is to design an automatic checkout system for retail stores that can accurately recognize and count products as they move along a checkout conveyor belt. I have briefly described the top-performing methods for this track.

4.1. Enhancing Retail Checkout Accuracy through Domain Generalization and Learning without Forgetting

The proposed solution is a two-stage pipeline for detecting and recognizing products in a retail checkout as shown in Figure 2. The first stage uses an object detector to propose class-agnostic boxes for cropping products. These proposed boxes are then fed into a classifier to recognize the corresponding categories of the products. To avoid duplicated counting, tracking and label voting are applied and the region for tracking is limited to the center of the image. The timestamp of a tracklet is estimated as the mean of the start and end time that the product is detected.One of the main challenges in this task is the domain gap be-

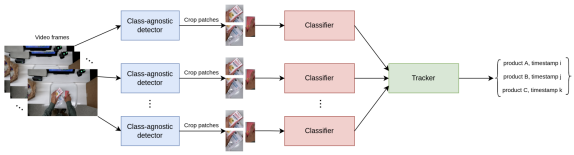

Figure 2. The proposed pipeline. In the first stage, a class-agnostic detector proposes boxes for cropping products. Then, a classifier categorizes corresponding classes of patches. Lastly, a tracker link predictions across frames to form the final result. [14]

Figure 3. The proposed pipeline. In the first stage, a class-agnostic detector proposes boxes for cropping products. Then, a classifier categorizes corresponding classes of patches. Lastly, a tracker link predictions across frames to form the final result. [14]

tween the synthetic data used to train the models and the real-world images used in testing.To address this challenge, the proposed method adopts several methods such as data generation, learning without forgetting, and model ensemble. These methods help to improve the domain generalization and robustness of the models. The data generation method consists of replacing various backgrounds for the objects, including plain color and synthesized images by GAN models, and adding objects from sources such as ShapeNet to reduce the false-positive rate. The learning without-forgetting method is used to preserve the generalization of pretrained models during finetuning and acts as a regularizer to reduce the domain bias towards the synthetic data. The model ensemble method is used to enhance the robustness of the classifier. In the preliminary experiments, a synthetic dataset is generated to train the detector. The synthetic dataset is created by generating various background images and pasting objects from sources such as the competition and ShapeNet datasets. The background images are generated using color randomization and GAN models. The foreground objects are augmented using methods such as brightness adjustment, horizontal/vertical flip, and blur to make them more diverse in appearance. To further improve the performance of the detector, the proposed solution adopts domain generalization methods and Learning without Forgetting (LWOF).LWOF is a method originally used for classification tasks and is applied to the object detection problem in this solution. The method has the advantage that it allows the model to remember the features learned from the old task while learning the new task. This is particularly useful in this scenario as the model is not allowed to use any external data except the synthetic one. In this way, LWOF helps to reduce the domain bias towards the synthetic data and acts as a regularizer to improve the performance of the detector on real-world images. In LWOF, the same image is fed to two models: a teacher and a student. The teacher is a pre-trained model that is frozen during training, while the student is initialized from the teacher but contains two classification heads. The first head is the regularization head, which is the same as the teacher's head, and the second one is a classification head for learning the new task. The classification outputs from the teacher and student regularization head are enforced to be similar by a similarity loss. This ensures that the model does not forget the old learned features, while the student classification head is supervised by the ground-truth label of the new task to learn the new patterns. After training, the teacher model and regularization head are removed. The Kullback-Leibler divergence loss is used to constrain the similarity between teacher and student outputs.

$$L(yst, yte) = yst \log \frac{yst}{yte} \tag{1}$$

This method is illustrated in 3 where the mask head and regression head are omitted to highlight the key idea of LWOF. The LWOF method is used to preserve the features learned from the old task while learning the new task. This helps to reduce the domain bias towards the synthetic data and acts as a regularizer to improve the performance of the detector on real-world images. The proposed method for object tracking and labeling involves linking detected objects and then determining the label for each track through a label voting process. Given a track T = t1, t2, ..., tN with N items, each item ti is categorized as class li. The label for the track is decided by computing the contribution score ci for each item ti:

$$c_i = \frac{f_\alpha(l_i)}{\sum_{j=1}^{N} f_\alpha(l_j)} \tag{2}$$

where $f_\alpha(l_i)$ is the frequency that class li appears in the video, a_i and s_i are the bounding box area and detection score of the item ti, and τ is the softmax temperature factor. Additionally, α, β, and γ are tuneable hyper-parameters. The label representing the track is then selected as the class

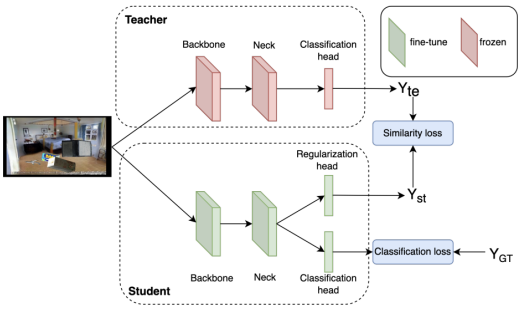

Figure 4. Illustration of the LWOF-style training for the detector. [15]

li corresponding to the highest contribution score ci. In terms of the classifier, the real dataset provided by the competition is used. The method also employs an ensemble of different backbones to enhance the robustness of the classifier. The final output of the pipeline is evaluated on the AI City challenge 2022 - Track 4 and achieves an F1 score of 40% on the test A set.

The proposed method as shown in Figure 2, first is an object detector purpose agnostic boxes for crop products. Then, these patches are fed into a classifier to recognize corresponding categories. To avoid duplicated counting, they apply tracking and perform label voting. they have also limited the region for tracking in the center of the image. Finally, the timestamp of a tracklet is estimated as the mean of the start and end time that the product is detected. They first generated the synthetic dataset for training detectors. They have computed the mean Average-Precision (mAP) score of the Mask-RCNN on our generated dataset and sorted values to take the top-score classes. Afterward, they use boxes of these 6 classes to crop patches for the following classifier. Regarding the classifier, they train the `RepVGG-A0` [6] for 10 epochs and obtain 99.99% top-1 accuracy. Finally, we cascade the detector and classifier to infer the test-A of the challenge and get F1 = 0.2769. The next part was data generation, where they employ background generation and foreground copy-paste. Concretely, the former consists of color randomization and synthesizing images by GAN models. Whereas, the latter copy objects from sources, including competition and ShapeNet datasets, then paste them into the background. The former is to diversify background concepts that help model robust to the real scene, whilst the latter is expected to reduce the false-positive rate. Lastly, they have adopted Learning Without Forgetting [10] which was originally used for the classification task, to our object detection problem. This method has the advantage that it can remember features learned from the old task when learning the new task. In LWOF, we feed the same image to two models: teacher and student. The teacher is a pretrained model and frozen during training, while the student is initialized from the teacher, but contains two classification heads: the regularization head which is the same as the teacher's head, and another a classification head for learning the new task. Classification outputs from the teacher and student regularization head are enforced to be closed by a similarity loss. They achieved `F1` score of 0.40 on Test Set A for AI City Challange Track4.

4.2. Image Inpainting

In [1], Bartl *et al*. presents a method for completing missing regions in an image, known as image inpainting. This task has various practical applications such as object removal, image retargeting, image compositing, and 3D photo effects. In recent years, Generative Adversarial Networks (GANs) have become a leading method for image inpainting, producing high quality and precise results. GANs consist of two main components, a generator for image synthesis and a discriminator for evaluation. In addition, two-stage networks can predict intermediate representations of an image such as edges, gradients, segmentation maps, or smoother images for final output enhancement. To improve the performance of GANs, various methods have been proposed such as incorporating dilated convolutions, designing contextual attention, using style code modulation, Fourier convolution, perceptual loss, and feature gating. The paper also presents a method for removing a person from the image which significantly reduced the false positive detection rate by using instance segmentation methods. In this process as they used synthetic images to train the model, so they inserted the object into "free space", and therefore they were isolated in the frames during training, and there were no other objects near the annotated products. For this reason, it has often been the case during inference that the worker's hands or body are detected as products even in cases where no product occurs in the scene, to avoid this problem they have proposed the method of image inpainting to remove the person, which significantly reduced the false positive detection rate. The method used to "delete" a person is LaMa [19]. This method requires an image and a related mask as its input. Thus, it is necessary to detect the person's mask as the first step. They have used the instance segmentation method from the MMDetection toolbox [5]. In particular, we tested the models DetectoRS [16], HTC [4], PointRend [9], and YOLACT [3]. All of these methods suffer from some inaccuracies, so they have combined all the methods. The inpainting is processed frame-by-frame. In order to identify the right object, they had to make sure that they have removed the unwanted area other than the object, for this they used localization of ROI

(region of interest) made automatically; the first step in this process is to extract the background image. This image for each scene is extracted by the following: Gaussian Mixture Model [21] extracts background from the part of each frame of the video sequence (with the usage of previous frames), and the mean value of all these background images is computed as the resulting background model. For the computation, inpainted video frames are used, as it makes the process easier by removing unnecessary objects (persons). When the mean background image is extracted for the scene, the image is transformed to grayscale, and the Scharr operator (an enhanced variant of the Sobel operator) for edge detection is applied. The Scharr operator is applied in the x and y direction and combined together (resulting edge detection is in Figure 11 bottom left). A flood fill algorithm is used on this image with detected edges, which searches the same/similar values as a seed (in our case, the seed is in the image center, but can be set arbitrarily). In this way, all pixels until edges are connected and marked as the "tray". The detected ROI can be seen in Figure 11 bottom right. When ROI is detected, it serves for filtering detections outside (with some extension). The resulting bounding box is an axis-aligned rectangle of all pixels found by the flood fill algorithm. This detector is trained on the synthetic dataset described in Section 3.2 and fine-tuned on the inpainted frames. The YOLOX detector is able to locate and classify the products in the scene, with the added benefit of handling multiple scales and aspect ratios. The detections are then tracked using a combination of object tracking and label voting to avoid duplicated counting. The timestamp of a tracklet is estimated as the mean of the start and end time that the product is detected. This allows for a more accurate representation of the products in the scene. Overall, the proposed method for product detection and tracking is able to achieve high precision and recall, making it a strong performer in the Ai City Challenge 2022 for Track 4.

The next step after ROI detection is the detection of the products themselves. For detection, they have used a multi-class YOLOX [7] detector, which reaches high precision on public datasets evaluation together with fast inference speed. They have trained their model with 116 output classes on the provided dataset, they achieved AP 97.47% on the validation set during training. As a part of their proposed model, they tried two tracking algorithms — SORT [2] and ByteTrack [20]. Both algorithms work online with bounding rectangles of the detections and use the Kalman filter to predict the future positions and merge these detections/predictions to corresponding tracks. The last step in their pipeline is the post-processing of the detected tracks. Each track t^{i} contains detections d_j^i.Each detection d_j^i is composed of a bounding box, class ID, class confidence, and detection confidence as was mentioned previously. As a track can contain certain inaccurate class values(in one track t^i can be detections d_j^i with different class IDs).

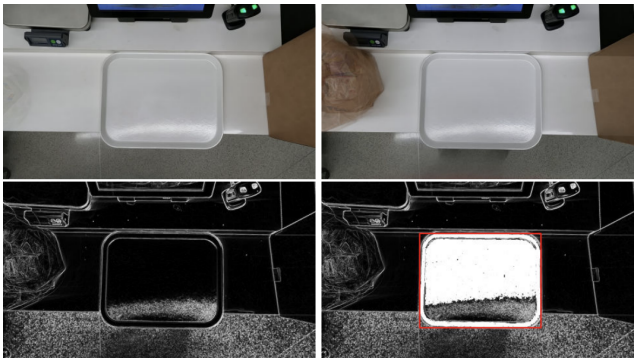

Figure 5. Top row: Mean background images for two sample scenes. Bottom left: Edges detection by the Scharr operator. Bottom right: Detection of the "tray" by the usage of flood fill algorithm; bounding rectangle in red.

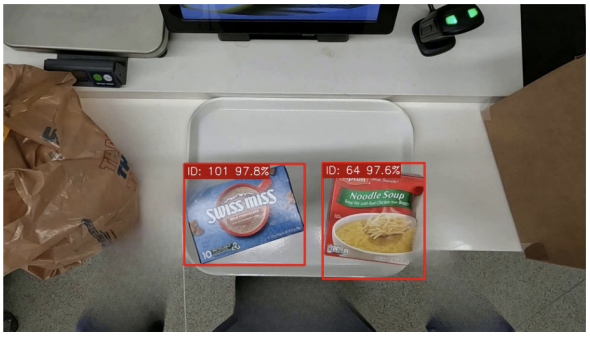

Figure 6. Multi-class detection with our trained YOLOX detector.

Figure 7. Sample of the synthetic dataset of products (top row) with generated object masks (bottom row).

4.3. Robust Automatic Checkout System using Deep Learning (RACS-DL)

The DeepACO system, which includes a detector, tracker, and counter, uses deep learning-based detectors and trackers and allows for different object detection models to be interchangeable. It also includes a hand estimation module that performs precise keypoint localization of hand-knuckle coordinates and is robust to partially visible

hands and self-occlusions. The system is implemented with both synchronous and asynchronous processing pipelines. The proposed DeepACO system includes a hand estimation module which is one of the major contributions to the system. This module is responsible for identifying hand-handled products by customers, and it uses the MediaPipe framework consisting of multiple models working together to achieve this goal. The models include a palm detection model that operates on the entire image and returns an oriented hand bounding box, and a hand landmark model that runs on the cropped image region defined by the palm detector and produces high-fidelity 3D hand key points. The palm detector is trained using the SSD model and is able to detect palms even in two-hand self-occlusion cases, like handshakes. The encoder-decoder feature extractor is used for bigger scene context awareness, even for small objects. The focal loss is minimized during training to support many anchors resulting from the high-scale variance. An average precision of 95.7% in palm detection can be achieved with these techniques. The hand estimation module is trained for 50 epochs with Stochastic Gradient Descent (SGD) and the best weights evaluated on the test set are selected for inference. The DeepACO system includes a tracking module, which assigns a unique id to a detected object when it enters the ROI and associates the currently tracking objects with newly detected objects, maintaining the unique id when moving through the camera. The SORT method is used as the online multi-object tracker, which uses Kalman Filter for motion prediction and the Hungarian algorithm for tracking assignment. The Kalman filter uses the unmatched detection results to initialize the tracking state as a new target and the matched detection results to update the existing target's tracking state. The state-space in each target is defined in the dimensional state space (u,v,s,r,u',v',s'), where u and v stand for the horizontal and vertical pixel 2D location of the center of the target. The scale s and r represent the scale (area) and the aspect ratio of the object's bounding box. The tracking states are defined as follows: New Track, Candidate Track, Confirmed Track, Counting Track, Counted Track, Exiting Track, and Deleted Track. Track management controls the object's existence in the whole program. The extension of the original track management of SORT is one of the significant contributions to the proposed DeepACO system.

4.4. Vision Transformer enhanced by U-Net and Image Colorfulness Frame Filtration for Automatic Retail Checkout

Shihab *et al*. [18] presents a multi-task product counting and recognition (MPCR) is a computer vision task that aims to identify product items from a collection of 116 classes using an image or video frame. The goal is to accurately detect and identify products in an image or video stream, for

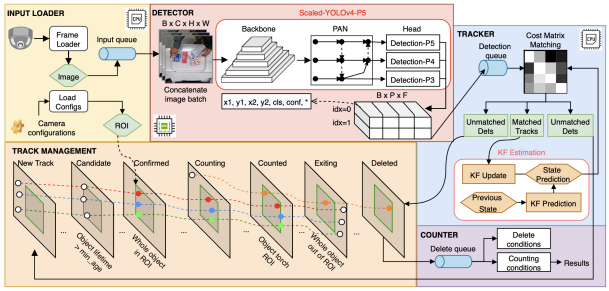

Figure 8. Sampled checkout images of three clutter levels.

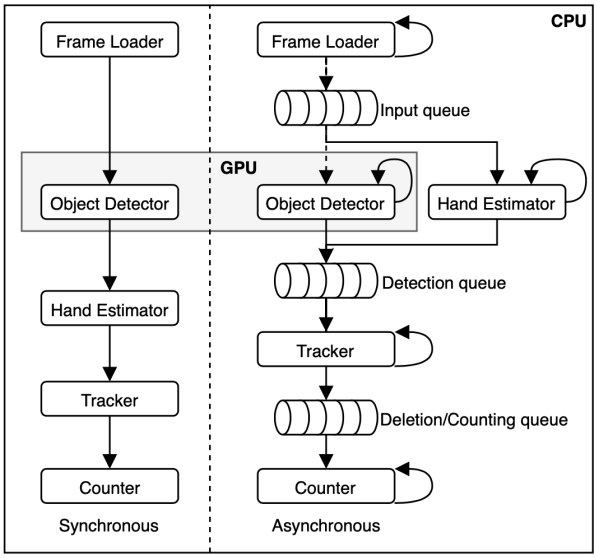

Figure 9. Synchronous and asynchronous processing pipelines are implemented. The synchronous pipeline is mainly used for developing and debugging the system, while the asynchronous pipeline is used for the final solution.

example, in a retail environment. To accomplish this task, a dataset of synthetic images is provided, each containing a single product item along with a binary segmentation mask indicating the region of interest (ROI) and the name of the product item. This dataset is used to train a model that can then be applied to real-world images or videos to detect and classify products.

$$dst(x,y) = \begin{cases} 0 & \text{if } src(x,y) > thresh \\ \text{maxval} & \text{otherwise} \end{cases} \quad (3)$$

The proposed method for MPCR utilizes the U-Net architecture, which is a convolutional neural network designed for image segmentation tasks. The U-Net architecture consists of a contracting path that captures context and a symmetrically expanding path that enables precise localization. The output of the model is a binary mask that shows

the class of each pixel in the image, indicating which pixels belong to a product and which do not.

To address the domain bias problem, which occurs when the training data is not representative of the real-world data that the model will be applied, the proposed method also uses a hand segmentation model. This model, which is a pre-trained DeepLabV3 with a ResNet-50 backbone, is used to remove hand regions from the image, as these regions are not present in the synthetic training data but are present in real-world images and videos. Furthermore, to remove foreign objects from the frames such as trays and bags, the method uses entropy masking. This technique segregates objects by using the textural cues in the image, it computes the entropy and binarizes the entropy image resulting in a mask that can be used to remove unwanted objects.

Finally, the Multi-Class Classification stage uses a Vision Transformer (ViT) architecture. ViT is a type of transformer-based neural network that has recently achieved state-of-the-art results in a variety of computer vision tasks. The proposed method has been tested and evaluated using this architecture, and it consistently performed well throughout the hyperparameter tuning process.The authors of this paper present a segmentation system that utilizes a unified single product item and hand segmentation stage, followed by entropy masking to address the issue of domain bias. The system then employs a Vision Transformer (ViT) for classification. The paper also explores the effects of various metrics to identify the best frames from a diverse test set, to accurately identify the region of interest and remove unnecessary noise from the frames. To achieve this, the authors propose a new metric, called the CBT metric, which is specifically designed for this dataset. The best method yielded by this system achieved the 3rd place in the AI City Challenge 2022 Track 4. The authors suggest that for future work, they aim to exploit the temporal information in video frames for improved identification of product items. Additionally, they plan to continue experimenting with other architectures and hyperparameters, as well as augmentation techniques.

5. Conclusion

This paper addresses the broad area of product recognition technologies. Product recognition will become increasingly important in a world where cost margins are becoming increasingly tight, and customers have increasing pressures on their available time. By summarising the literature in the field, we make research in this area more accessible to new researchers, allowing for the field to progress. It is very important that this field addresses these four challenging problems: (1) large-scale classification; (2) data limitations; (3) intraclass variation; and (4) flexibility. We have identified several areas for further research: (1) generating data with deep neural networks; (2) graph neural networks with

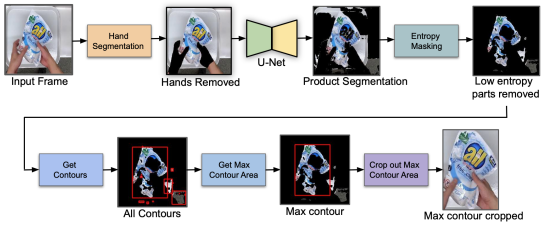

Figure 10. Schematic layout of the U-Net-based segmentation and contour selection stage. Given an optionally filtered input image, we first segment the hand to address the domain bias problem. The product segmentation is done followed by entropy masking to get the final ROI. After this, contours are located and the maximum contour is kept to reduce noise, which is then cropped from the final image. Finally, the cropped image is fed to ViT, for the final classification of the product item.

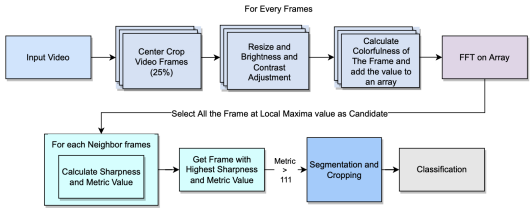

Figure 11. Overview of Test Video Preprocessing and Frame selection stages.

deep learning; (3) cross-domain recognition with transfer learning; (4) joint feature learning from text information on the packaging; (5) incremental learning with the CNN; and (6) the regression-based object detection methods for retail product recognition.

In this article, we have presented an extensive review of recent research on deep learning-based retail product recognition, with more than one hundred references. We propose four challenging problems and provide corresponding techniques to those challenges. We have also briefly described the publicly available datasets and listed their detailed information, respectively.

Overall, this paper provides a clear overview of the current research status in this field and that it encourages new researchers to join this field and complete extensive research in this area.

References

[1] Vojtěch Bartl, Jakub Špaňhel, and Adam Herout. Persongone: Image inpainting for automated checkout solution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 3115–3123, June 2022. 4

[2] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple online and realtime tracking. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 3464–3468, 2016. 5

[3] Daniel Bolya, Chong Zhou, Fanyi Xiao, and Yong Jae Lee. Yolact: Real-time instance segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019. 4

[4] Kai Chen, Jiangmiao Pang, Jiaqi Wang, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. Hybrid task cascade for instance segmentation, 2019. 4

[5] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, and Xu. Mmdetection: Open mmlab detection toolbox and benchmark, 2019. 4

[6] Xiaohan Ding, Xiangyu Zhang, Ningning Ma, Jungong Han, Guiguang Ding, and Jian Sun. Repvgg: Making vgg-style convnets great again. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13733–13742, June 2021. 4

[7] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. Yolox: Exceeding yolo series in 2021, 2021. 5

[8] Marian George and Christian Floerkemeier. Recognizing products: A per-exemplar multi-label image classification approach. In *European Conference on Computer Vision*, pages 440–455. Springer, 2014. 2

[9] Alexander Kirillov, Yuxin Wu, Kaiming He, and Ross Girshick. Pointrend: Image segmentation as rendering, 2019. 4

[10] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2935–2947, 2018. 4

[11] Diego López-de Ipiña, Tania Lorido, and Unai López. Indoor navigation and product recognition for blind people assisted shopping. In *International Workshop on Ambient Assisted Living*, pages 33–40. Springer, 2011. 2

[12] Fumikazu Morimura and Kenichi Nishioka. Waiting in exit-stage operations: expectation for self-checkout systems and overall satisfaction. *Journal of Marketing Channels*, 23(4):241–254, 2016. 2

[13] Milind Naphade, Shuo Wang, David C. Anastasiu, Zheng Tang, Ming-Ching Chang, Yue Yao, Liang Zheng, Mohammed Shaiqur Rahman, Archana Venkatachalapathy, Anuj Sharma, Qi Feng, Vitaly Ablavsky, Stan Sclaroff, Pranamesh Chakraborty, Alice Li, Shangru Li, and Rama Chellappa. The 6th ai city challenge, 2022. 2

[14] Milind Naphade, Shuo Wang, David C. Anastasiu, Zheng Tang, Ming-Ching Chang, Yue Yao, Liang Zheng, Mohammed Shaiqur Rahman, Archana Venkatachalapathy, Anuj Sharma, Qi Feng, Vitaly Ablavsky, Stan Sclaroff, Pranamesh Chakraborty, Alice Li, Shangru Li, and Rama Chellappa. The 6th ai city challenge, 2022. 2, 3

[15] Thuy C. Nguyen, Nam LH. Phan, and Son T. Nguyen. Improving domain generalization by learning without forgetting: Application in retail checkout. 4

[16] Siyuan Qiao, Liang-Chieh Chen, and Alan Yuille. Detectors: Detecting objects with recursive feature pyramid and switchable atrous convolution, 2020. 4

[17] Bikash Santra and Dipti Prasad Mukherjee. A comprehensive survey on computer vision based approaches for automatic identification of products in retail store. *Image Vis. Comput.*, 86:45–63, 2019. 1

[18] Md. Istiak Hossain Shihab, Nazia Tasnim, Hasib Zunair, Labiba Kanij Rupty, and Nabeel Mohammed. Vista: Vision transformer enhanced by u-net and image colorfulness frame filtration for automatic retail checkout, 2022. 6

[19] Roman Suvorov, Elizaveta Logacheva, and Mashikhin. Resolution-robust large mask inpainting with fourier convolutions, 2021. 4

[20] Yifu Zhang, Peize Sun, and Yi Jiang. Bytetrack: Multi-object tracking by associating every detection box, 2021. 5

[21] Zoran Zivkovic and Ferdinand van der Heijden. Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern Recognition Letters*, 2006. 5