DOKUZ EYLÜL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

DEVELOPMENT OF INTELLIGENT SYSTEMS USING AUGMENTED REALITY AND MACHINE LEARNING TECHNIQUES

by
Ramiz YILMAZER

February, 2021
İZMİR

DEVELOPMENT OF INTELLIGENT SYSTEMS USING AUGMENTED REALITY AND MACHINE LEARNING TECHNIQUES

A Thesis Submitted to the
Graduate School of Natural and Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy in Computer Engineering

by
Ramiz YILMAZER

February, 2021
İZMİR

Ph.D. THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"DEVELOPMENT OF INTELLIGENT SYSTEMS USING AUGMENTED REALITY AND MACHINE LEARNING TECHNIQUES"** completed by **RAMİZ YILMAZER** under supervision of **ASSOC. PROF. DR. DERYA BİRANT** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Doctor of Philosophy.

Assoc. Prof. Dr. Derya BİRANT

Supervisor

Prof. Dr. Alp KUT	Asst. Prof. Dr. Pelin YILDIRIM TAŞER
Thesis Committee Member	Thesis Committee Member
Assoc. Prof. Dr. Aytuğ ONAN	Assoc. Prof. Dr. Özgü CAN
Examining Committee Member	Examining Committee Member

Prof. Dr. Özgür ÖZÇELİK

Director

Graduate School of Natural and Applied Sciences

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to my supervisor, Assoc. Prof. Dr. Derya BİRANT, for her immense knowledge, continuous support, great mentorship, encouragement, and useful suggestions throughout this study. Her guidance helped me in all the time of my research and writing of this thesis. It has been an honor to be her Ph.D. student.

I am also thankful to the rest of my thesis committee: Prof. Dr. Alp KUT, and Asst. Prof. Dr. Pelin YILDIRIM TAŞER for their valuable advice for this study.

Besides, I would like to thank my family for their support and encouragement.

Ramiz YILMAZER

DEVELOPMENT OF INTELLIGENT SYSTEMS USING AUGMENTED REALITY AND MACHINE LEARNING TECHNIQUES

ABSTRACT

The aim of the thesis is to develop intelligent systems using augmented reality and machine learning techniques. Particularly, we focused on managing and checking on-shelf availability (OSA) in the retail sector since providing high OSA is a key factor to increase profits in grocery stores. Recently, there has been a growing interest in computer vision approaches to monitor OSA. However, the large and well-known computer vision datasets do not provide annotation for store products, and therefore a huge effort is needed to manually label products on images. To tackle the annotation problem, this thesis proposes a new method.

Our study combines two concepts "semi-supervised learning" and "on-shelf availability" (SOSA) for the first time. Moreover, it is the first time that “You Only Look Once” (YOLOv4) deep learning architecture is used to monitor OSA. Furthermore, this thesis provides the first demonstration of explainable artificial intelligence (XAI) on OSA. It presents a new software application, called SOSA XAI, with its capabilities and advantages. In addition, a new augmented reality (AR) application was developed to monitor the latest status of the shelf, called SOSA AR.

In the experimental studies, the effectiveness of the proposed SOSA method was verified on a real-world image dataset, with different ratios of labeled samples varying from 20% to 80%. The experimental results show that the proposed approach outperforms the existing approaches (RetinaNet and YOLOv3) in terms of accuracy.

Keywords: On-shelf availability, semi-supervised learning, deep learning, image classification, machine learning, explainable artificial intelligence, augmented reality

ARTIRILMIŞ GERÇEKLİK VE MAKİNE ÖĞRENMESİ TEKNİKLERİ KULLANILARAK AKILLI SİSTEMLERİN GELİŞTİRİLMESİ

ÖZ

Tezin amacı, artırılmış gerçeklik ve makine öğrenmesi tekniklerini kullanarak akıllı sistemler geliştirmektir. Özellikle, perakende sektöründe rafta bulunabilirliği (OSA) yönetmeye ve kontrol etmeye odaklanılmıştır çünkü OSA'yı yüksek seviyede tutmak marketlerde karı artırmak için önemli bir faktördür. Son zamanlarda, OSA'yı izlemek için bilgisayarlı görü yaklaşımlarına artan bir ilgi vardır. Ancak, büyük ve iyi bilinen bilgisayarlı görü veri setleri, market ürünleri için etiketlenmiş resimler sunmamaktadır ve bu nedenle, ürünleri resimler üzerinde manuel olarak etiketlemek için büyük çaba gereklidir. Bu problemin üstesinden gelebilmek için tezde yeni bir yöntem önerilmektedir.

Çalışmamız iki kavramı "yarı denetimli öğrenme" ve "rafta bulunabilirlik" (SOSA) ilk kez birleştirmektedir. Ayrıca, OSA'yı izlemek için "You Only Look Once" (YOLOv4) derin öğrenme mimarisi ilk kez kullanılmaktadır. İlave olarak, bu tez ile açıklanabilir yapay zeka (XAI) OSA üzerinde ilk kez uygulanmıştır. Bu kapsamda geliştirilen SOSA XAI adında yeni bir yazılım uygulaması yetenekleri ve avantajları ile birlikte sunulmaktadır. Ayrıca rafın son durumunu izlemek için SOSA AR adı verilen yeni bir artırılmış gerçeklik (AR) uygulaması geliştirilmiştir.

Deneysel çalışmalarda, önerilen SOSA yönteminin etkinliği, %20 ile %80 arasında değişen oranlarda etiketlenmiş resimler içeren gerçek bir veri seti üzerinde doğrulanmıştır. Deneysel sonuçlar, önerilen yaklaşımın doğruluk açısından mevcut yaklaşımlardan (RetinaNet ve YOLOv3) daha iyi performansa sahip olduğunu göstermektedir.

Anahtar kelimeler: Rafta bulunabilirlik, yarı denetimli öğrenme, derin öğrenme, resim sınıflandırma, makine öğrenmesi, açıklanabilir yapay zeka, artırılmış gerçeklik

CONTENTS

Page

LIST OF FIGURES

Page

LIST OF TABLES

Page

CHAPTER ONE
INTRODUCTION

1.1 General

Machine learning techniques have been applied to different areas in the retail sector. One of them is monitoring *on-shelf availability* (OSA) in grocery stores. Providing high OSA is a key factor to increase profits. When a product that a shopper looks for is not available on its designed shelf, also known as "*out-of-stock*" (OOS), this causes a negative impact on the customer behaviors in the future.

According to the research reported by Corsten & Gruen (2003), when a product is not available on the designed shelf space, 31% of consumers buy the product from a different store, 26% of them buy a different brand, 19% of them buy a different size of the same brand, 15% of them buy the same product at a later time, and 9% of them buy nothing. Besides, another study by Musalem, Olivares, Bradlow, Terwiesch, & Corsten (2010) showed that the "out of the stocks" rate is about 8% in the United States and Europe. For this reason, OSA has a significant effect on business profit. The remaining products can be checked using an inventory management system, but it only shows the number of products in the stock. These products, available in stock, might not be on the shelves. Current inventory systems cannot understand the number of products on the shelves. OSA is checked by employees manually at most of the grocery store. This approach is not effective and sustainable since it continuously requires human effort. For this reason, a new intelligent system is proposed in this thesis.

There are several studies to automate monitoring OSA. These studies consider the subject from different perspectives. One of them proposes Radio Frequency Identification (RFID) tagging to monitor product quantity on the shelves (Bottani, Bertolini, Rizzi, & Romagnoli, 2017), but this approach is not cost-effective to implement the technology and integrate it into existing systems (Michael & McCathie, 2005). Some of them applied traditional image processing techniques to detect the presence and absence of the product such as (Moorthy, Behera, Verma, Bhargave, &

Ramanathan, 2015), and some of them used deep learning approaches for object detection on the shelves such as (Higa & Iwamoto, 2018). When all these previous works are examined, there are pros and cons of these approaches. When traditional image processing methods have been used such as Histogram of Oriented Gradient (HoG) for feature extraction and Support Vector Machine (SVM) for a classifier, they have limited performance even if on the large datasets and hard to be increased their performances. In addition, the visual similarity among the different products of the same brand can lead to misclassification. On the other hand, when deep learning (DL) approaches were used such as Recurrent Convolutional Neural Network (RCNN) or You Only Look Once (YOLO), high accuracies could be achieved.

Semi-supervised learning is a type of machine learning paradigm that aims to build a classifier from both labeled and unlabeled data. A semi-supervised learning approach can produce results with satisfactory accuracy using an amount of labeled data and a much larger amount of unlabeled data in the training phase (Zhu, 2017).

1.2 Purpose

On-shelf availability is a key performance indicator of how available a product is for the shoppers in a grocery store. The products should be in the shelf where a shopper expects it. To ensure a high on-shelf availability, store clerks should move inside the store and check the products at the shelves. Nevertheless, this task is quite time-consuming since there are usually hundreds or thousands of different products to keep track of in a grocery store. Therefore, this thesis aims to simplify this task for store clerks by developing an intelligent system that can automatically identify “out of stocks” products.

In the OSA applications, the deep learning algorithms require annotated shelf images to train, but the largest and well-known computer vision datasets do not provide annotation for store products, and therefore a huge effort is needed to manually label products on images. This thesis aims to reduce labeling cost and provide additional knowledge present in unlabeled data, by using semi-supervised learning. For this

purpose, it proposes a new method that combines two concepts "semi-supervised learning" and "on-shelf availability" (SOSA) for the first time. An important advantage of the proposed SOSA method is that it solves the OSA problems where labeled image data is scarce. Labeling OSA image data is an expensive, tedious, difficult, or time-consuming process since it requires human labor. The proposed SOSA method deals with the design of on-shelf availability monitoring models in the presence of both labeled and unlabeled image data.

From another perspective, users have to understand and trust to the constructed OSA model as a kind of artificial intelligence (AI) application. Besides, if their requirements change, they have to be managed these applications according to their needs (Barredo Arrieta et al., 2020). Therefore, this thesis aims to enable users to manage, understand, and trust the OSA model by using explainable artificial intelligence (XAI) techniques.

Another purpose of this thesis is to develop an augmented reality (AR) application to monitor the last statuses of the shelves. To this aim, we implemented a new mobile application, called SOSA AR. Using this application, store clerks can easily monitor notifications and last audits when they working on shelves.

1.3 Main Contributions of the Thesis

The main contributions and novelty of this thesis can be listed as follows. (1) This is the first study that combines two concepts "semi-supervised learning" and "on-shelf availability" (SOSA) for the first time. (2) It is the first time that YOLOv4 deep learning architecture is used for exploring OSA. (3) This study is also original in that it compares three deep learning approaches for monitoring OSA in the retail sector. (4) To the best of our knowledge, this is the first demonstration of Explainable AI on OSA. This thesis also presents a new software application, called SOSA XAI, with its capabilities and advantages. (5) An augmented reality (AR) mobile application is developed to monitor the latest status of the shelf, called SOSA AR.

In the experimental studies, the effectiveness of the proposed SOSA method was verified on image datasets, with different ratios of labeled samples varying from 20% to 80%. We focused on category-based detection of empty and almost empty shelves. For this purpose, we used both labeled and unlabeled images that include 49573 products grouped under three categories. The experimental results show that the proposed approach outperforms the existing approaches (RetinaNet and YOLOv3) in terms of accuracy.

The proposed SOSA method and the SOSA XAI approach were introduced in (Yilmazer & Birant, 2021b) , while the proposed SOSA AR application was explained in (Yilmazer & Birant, 2021a).

1.4 Organization of the Thesis

The thesis consists of seven chapters. The remainder of this thesis is structured as follows:

In Chapter 2, previous studies about object detection are summarized. In addition, object detection approaches, specialized to monitor OSA, are presented.

In Chapter 3, a summary is given about deep learning. Furthermore, two-stage detectors are explained and brief information about three two-stage detector algorithms (R-CNN, Faster R-CNN, FPN) is given. Moreover, one-stage detectors are explained and three one-stage detector algorithms (RetinaNet, YOLOv3, YOLOv4) are described in detail.

In Chapter 4, a brief summary is given about deep learning that is specialized to object detection. Also, our proposed approach (Semi-Supervised Learning on OSA - SOSA) is explained, and our developed systems (Explainable AI for OSA (XAI SOSA) and Augmented Reality Application for OSA (SOSA AR)) are demonstrated.

In Chapter 5, the dataset used in this study is firstly described, and then, experimental studies and results are presented.

In Chapter 6, discussions are given in two main aspects: the main findings of this study and the challenges that could be encountered and overcome.

Finally, concluding remarks and possible future works are presented in Chapter 7.

CHAPTER TWO
LITERATURE REVIEW

In the literature, researchers have been approached the subject from different perspectives. They proposed both technical and managerial solutions to minimize OOS and monitor OSA. Different technical solutions have been proposed such as analyzing textual data on the relational database management systems (RDMS) and applying computer vision-based techniques. In this thesis, we focus on computer vision-based techniques, and previous works were explained based on this scope.

2.1 Review of Previous Studies

Internet of Things (IoT) based applications were proposed for OSA in several studies. Vargheese & Dahir (2014) proposed architecture using IoT elements such as Illumination Source (IR), sensors, and cameras with PTZ capability. Their proposed method aimed to improve operational efficiency and customer experiences. For retail OSA management, sensors' data were collected and analyzed in multiple retail scenarios. Multiple methods were used to identify the presence/absence of a product. Firstly, a sensor was placed at the back of the shelf. The aim of the sensor was that, checking at least one product was existent on the shelf. Another method was the top-down model. IR sensor was placed at the top of the shelf to count the number of products. Thirdly, a camera with PTZ (pan, tilt, zoom) capability was placed in front of the shelf. Captured images that were taken from the camera were analyzed to identify products using computer vision algorithms. Their study does not give technical details about approaches, instead, it proposed a model based on how to improve efficiency for OSA.

Bottani, Bertolini, Rizzi, & Romagnoli (2017) studied a pilot project to manage OSA and out-of-stock (OOS) based on backroom and shelf for fresh foods. Radio Frequency Identification (RFID) technology was used in their project. Data was collected from RFID tags and these were evaluated by the developed modules. Also, key performance indicators were reported based on product traceability, product

history, product availability, and inventory. On the other hand, RFID technology has some pros and cons for supply chain management (SCM). Michael & McCathie (2005) examined these pros and cons. The main advantages of that RFID tags can be read from any direction but barcode tags need proper orientation to read using barcode readers. The technology provides to track inventory, assets, and so on. On the other hand, additional costs are needed to implement the technology and integrate it into existent systems. Also, RFID tags are expensive for some implementations. In most of the scenarios to track products, these tags can be used for one time. Another critical point is that privacy issues when using RFID tags. Protocols that are used in the technology are designed to provide an optimal and efficient connection between RFID readers and tags. Consumer privacy concerns were not focused on when designing the protocols.

Several datasets have been collected from retail shelves. Klasson, Zhang, & Kjellström (2019) collected fruit, vegetable, dairy, and juice products' images from related sections and refrigerated sections in 18 different grocery stores. Dataset has 5125 images in 81 classes. The class was designed as hierarchical. These images were captured using a 16-megapixel smartphone camera from different perspectives and lighting conditions. Also, the dataset was split as train and test. Moreover, some classification methods were proposed and used such as CNN and Support Vector Machine (SVM) on the collected dataset. These classification models were used to evaluate collected dataset performance.

Another dataset was collected by Jund, Abdo, Eitel, & Burgard (2016). The dataset has 4947 images in 25 classes. The image count is changed between 97 and 370 in classes. Images were collected from different groceries using four different smartphone cameras. Each class contains different packaging and brands. To evaluate the dataset, some deep neural network classifiers were used such as CaffeNet and AlexNet.

García-Arca, Prado-Prado, & González-Portela Garrido (2020) proposed a methodology for redesigning the store and logistics processes. It was aimed that

improving measurement of on-shelf efficiency and logistics management in the supply chain. Also, reducing OOS events were aimed. KPI was used to measure shelf management and they processed the study based on store design, demand data, re-stacking cycle, replenishment cycle, and packaging system.

Some researchers approach the issue through the management of stocks. Bayle, Laurent, & Macé (2006) examined out of stock events based on store size and store sales in the category. They analyzed the frequency of partial OOS and complete OOS in the stores. The study proposed a new measure of OOS with collected data. In this study, data was collected with store audits. The operation was processed with manual visual checking of shelves.

Satapathy, Prahlad, & Kaulgud (2015) proposed a solution using image processing techniques with a smartphone camera and camera in the retail stores for on-shelf availability. .NET Framework, AForge.NET, and OpenCV frameworks and libraries were used in the study. Developed software accepts a single image from a smartphone camera or camera in the store. Two image processing approach were implemented First one is that basic processing and comparison was implemented without process detail of images. Secondly, an exhaustive template comparison was implemented. Input image was compared with different reference templates to identify the product on the shelf. The first approach was faster and time-saving than the second one. On the other hand, the second one has more accurate results.

Cheng, Yang, & Zhou (2005) proposed an improved algorithm for background subtraction. Background subtraction is a well-known approach to detect object changes in the images. They enhanced the Gaussian mixture model (GMM) algorithm to adapt changes to the image. An appropriate number of components and parameters can be updated automatically by the improved algorithm. This operation is repeated for each pixel on the image. Also, the processing time is reduced with the algorithm.

A deep neural network approach was proposed to detect image changes from the difference image by Zhao, Gong, Liu, & Jiao (2014). Three different datasets were

used in the study. In the algorithm, synthetic aperture radar (SAR) images were used as input. Two images, taken from the same area in a different time period, were given to the algorithm. The difference image was generated from these input images. The pre-classification operation was applied to the difference image. After these operations, a deep neural network was trained and fine-tuned. End of these operations, the ending condition was checked. Some evaluation metrics were used such as overall error, percentage correct classification, and kappa. These metrics' results were compared to different algorithms which were GKI and RFLICM.

Bay, Ess, Tuytelaars, & Van Gool (2008) studied detector and descriptor based on the Speed-Up Robust Features (SURF) algorithm. They focused on interest point detection, description, and matching. They worked on integral images to reduce computation time for interest point operations. Different approaches were used such as the Hessian matrix and interpolation. Also, it evaluated their proposed detector and descriptor. Moreover, they worked on the calibration of cameras using the SURF algorithm.

Some studies are existent to extract features from images using different techniques. Low (2004) studied extracting distinctive invariant features from images. Cascade filtering approach was used and operation cost was reduced in this way. Some stages were followed to extract features from images which were scale-space extrema detection, keypoint localization, orientation assignment, and keypoint descriptor. Different effective algorithms were selected and used for each stage. They worked on SIFT key points to match the image keypoint with the key point which is taken from the database. Also, these key points were used for object detection. The nearest-neighbor approach and the Hough transform approach were used.

Some studies focus on consumer behaviors based on retail out of stock situations. Gruen, Corsten, & Bharadwaj (2002) reported consumer behaviors. They examined 661 retail stores, 71,000 consumers, and addressed out of stock situations for goods in 32 categories. Research results were shared based on customer reactions when OOS conditions occur. %31 of customers purchases products from a different store. %26 of

customers purchases a different brand. %19 of customers purchases the different sizes of the same brand if existent. 15% of customers postpone the purchasing and %9 of customers does not purchase anything. Also, out of stock conditions were examined and addressed based on the day of week and time of time. Results showed that OOS was increased on Sunday and consumers preferred to go stores on the weekend and following days Monday and Tuesday also have a large rate for OOS. Moreover, OOS was increased in the evenings and studies showed that this situation was increased after 20:00.

Some studies exist for reducing OOS in retail stores. Gruen & Corsten (2008) prepared a guide about this. They examined that, cost relationship between reducing OOS operation and lost sales when OOS occurred. The study focused on store OOS and shelf OOS conditions. Each of these conditions needs different solutions to decrease OOS rates. Researchers focused on seven different root causes which were product item data accuracy, ordering and inventory accuracy, demand forecasting accuracy, store and shelf replenishment, shelf space allocation, planogram compliance, and item management.

Senior et al. (2007) studied on a tool to analyze videos that were taken from surveillance cameras for retail stores. They worked on object tracking and face tracking and the provided tool can be reached from a web browser. Collected metadata was stored as XML format in the database. Adaboost learning was used to detect faces. Also, Gabor wavelets were used for geometric structure from training samples. Color field tracker was used to track faces that were not clearly visible issues. The tool was developed to help loss prevention in retail stores. Also, they provide an event-based alarm to notify authorized staff for unauthorize entrance to restricted areas in the stores. Furthermore, the tool counts customers who were entering the store.

Morimoto et al. (2005) proposed an algorithm for object tracking in video pictures based on image segmentation and pattern matching. They detected all objects in the image and applied pattern matching. Their algorithms consist of object tracking, image segmentation, and motion determination. They compared objects in the previous frame

and current frame for object tracking and motion determination. Minimum distance search was used. Also, Euclidean and Manhattan distances were tested to evaluate tracking quality.

Angeles (2005) examined RFID technology, its applications, and its implementation samples. Also, guidelines and implementation strategies were explained. The study was focused on the supply chain field. Different RFID case studies and implementation details that were applied by different companies were explained. The first one is a smart pallet system to track pallet movements when products were on the pallet. RFID tag was attached for each pallet and RFID readers were installed on the warehouse bay doors. While pallets passed through these doors, they detected, tracked and pallets tag info was sent to software and stored in the database. The second one was implemented to manage the shipment of goods for the crate of a car manufacturer. The tag was attached for each crate and readers and related antennas were installed on the warehouse doors like the previous solution. Detected goods were checked shipment information that is taken from the database. If these taken shipment goods information was matched, a green light was fired and shipment was continued. Otherwise, a red light was fired and shipment was stopped. Another application was implemented to manage the food process and control the movement of raw materials for a food manufacturer. Also, some applications were implemented for inventory checking and management processes.

Some studies are examined out-of-stock situation from different perspectives such as goods-dominant (G-D) and service-dominant. Ehrenthal, Gruen, & Hofstetter (2014) examined out-of-stocks based on S-D logic. In G-D logic, direct sales losses are a fundamental point for OOS. On the other hand, S-D logic focuses overall benefits and costs based on rendered services between manufacturer, retailer, and shopper. Researchers gave a model to provide shelf availability. Also, they gave the formula to calculate the cost of OOS based on G-D and S-D logic. According to the study, a shopper can be purchased an item for another person. This third person was named as a user in the study. The shopper and the user can be the same person and different. This change caused different effects on behaviors when OOS occurred. In this phase,

they examined the OOS state from the manufacturer's side. The manufacturer could not track OOS states from retail stores directly but users can be changed brand type when they cannot find the item in the store. When this kind of situation is repeated, the manufacturer may have to be changed the production plan. Also, all these effects were examined in their study.

Papakiriakopoulos (2005) proposed an information system to detect OOS. It uses machine learning techniques based on a heuristic approach. A rule-based system was implemented. OOS situation was tried to catch using historical data of sales and ordering info. Information was taken from the retail information system. Their proposed system has two versions. The first version used 6 months period data and the second version used 6 months and 14 months period data with some improvements. The system has three modules which are the data consistency module, attribute measurement module, and inference module. It sends the detected OOS list to store managers and product suppliers. Their system detected %27 OOS issues daily with greater than %90 accuracy. Some of the OOS cases could not be caught by the system.

Bargoti & Underwood (2017) studied on fruit detection in orchards. A Faster R-CNN framework was proposed to accomplish the task. Images were collected from orchards and ImageNet dataset was used. They tried to detect three types of fruits which were apples, mangoes, and almonds. An architecture that was state-of-art was proposed for a faster R-CNN approach. Two networks, ZF and VGG16, were used for R-CNN. ZF network contains 5 convolutional layers and VGG16 network contains 13 convolutional layers. Results show that detection of small fruits was more error-prone and high-resolution images have to be used to increase detection accuracy and reduce false-positive rates.

Judith, Sigal, Mori, & Mandt (2018) examined using informed priors for deep representation learning. Some advantages were existing when using deep neural networks with Bayesian models. Models were designed to classify data. On the other hand, sometimes a semi-supervised learning approach has to be used but when data is not labeled completely, the designed structure has to be changed. In representation

learning, the same model can be used without changing. This can be achieved using informed priors with full covariance matrices. They showed how this operation is applied.

Donahue et al. (2014) studied deep convolutional activation features for generic visual recognition. They tried to extract generic visual features using their proposed approach. Also, pre-trained models were used for different dataset examples to get accurate results for the generic visual recognition task. The main point of the study is that, finding a way to create a generic CNN model and using the model for different datasets without any modification. Support vector machine and logistic regression classifiers were used to evaluate their results. Different datasets were used such as Caltech-101 the office domain adaptation dataset, Caltech-UCSD birds fine-grained recognition dataset, and SUN-397 scene recognition dataset.

Huang, Liu, Van Der Maaten, & Weinberger (2017) proposed state-of-start convolutional neural network architecture which is a dense convolutional neural network (DenseNet). It is similar to ResNet but it has some architectural differences. They tested their network using different datasets which are CIFAR, SVHN, and ImageNet. Test results showed that accuracy improvements with fewer parameters and less computation are possible in the proposed network.

Krizhevsky, Sutskever, & E. Hinton (2012) studied deep convolutional neural networks using large datasets which are subsets of ImageNet. ImageNet dataset has 15 million labeled images and they have subsets which are LSVRC-2010 AND ILSVRC-2012. ILSVRC-2010 have 1.2 million images and they trained their networks using these 1.2 million labeled images for classification purpose. Their designed network had 8 layers. 5 of them was convolutional layer and 3 of them fully connected layers. Also, max pooling and rectified linear unit (ReLU) approaches were used on the network. The designed network had 60 million parameters and 650,000 neurons. The training was completed end of the 6th day.

Mureşan & Oltean (2018) worked on a new dataset that contains fruit images and applied a neural network to classify images. The dataset contains 82213 images and 120 fruits and vegetables. They designed convolutional neural networks with different structures using the TensorFlow library which is an open-source library developed by Google. Some scenarios were experimented with based on image modifications such as RGB, HSV, grayscale, hue/saturation changes, and flips. Ten different types of CNN were designed and experimented with using different parameter settings.

Ngiam et al. (2011) proposed a deep learning application to learn features from multiple modalities which are audio and video samples. They used several datasets which are CUAVE, AVLetters, AVLetters2, Standford dataset, and Timit. They tried to find connected features between audio and visual data for speech recognition. Different multimodal learning settings were examined which are classic deep learning, multimodal fusion, cross-modality learning, and shared representation learning. Some of these approaches accept only one type of modalities such as audio or video and some of these accept multiple modalities such as both audio and video. They tried to pair lip movements to voice using some speaking datasets using different multimodal learning settings. Unlabeled data was used in the study to learn features.

Oquab, Bottou, Laptev, & Sivic (2014) worked on transferring weights from trained CNN to new CNN. That is to say, a dataset was used to train designed CNN, and some features and weights were extracted in this way. These weight transfer to new CNN to use on a new dataset. They worked on the ImageNet dataset as a source dataset and trained their CNN. Pascal VOC 2007 and 2012 datasets were used as target datasets. Classification results were tested on the target dataset. Transferring weights from source to target is a difficult task. Adaptation layers were added to target CNN to accomplish the issue. The critic point is that the pre-trained of the internals layers were transferred to the target network.

Radford, Metz, & Chintala (2016) studied deep convolutional generative adversarial networks (DCGANS) based on unsupervised learning. They tested their approach using three different datasets. Images were tried to generate from images that

were taken from datasets. LSUN, Imagenet-1k, and face datasets were used for this purpose. LSUN dataset contains bedroom images and it has over 3 million images. Using a designed network, images were generated after the first epoch. Generated images' quality was low. The Imagenet-1k dataset contains natural images for unsupervised learning and these images were used in training face. Also, the faces dataset contains 3 million face images for 10 thousand people and some interpolation operations were tried using the dataset. For instance, different features between smiling woman images and neutral woman images were found, and added these features to a neutral man. After this operation, a smiling man was seen in the produced images. Another example was that different features between a man with glasses images and a man without glasses images were found and added these features to a woman without glasses. After this operation, a woman with glasses was seen in the produced images.

The generative adversarial networks (GAN) approach was improved and implemented to detect an object from blur images with high accuracies. Prakash & Karam (2019) proposed the Single Shot MultiBox Detector based Detection of Objects (GAN-DO) framework for this purpose. They used Single Shot MultiBox Detector (SSD) and RetinaNet one-stage detectors and they tried to improve the result of accuracies taken from these models using their proposed approach. Also, Wang, Hong, Wang, & Yu (2020) proposed the GAN-Knowledge Distillation approach to increase one-stage detectors performance. For this purpose, they used teacher and student networks. The teacher network has deeper and larger architecture according to the student network. Generated features from the teacher network were used in the student network to increase the one-stage detector performance. In their study, the SSD one-stage detector was used to test their proposed approach.

Sa et al. (2016) proposed a novel system that uses deep convolutional neural networks (DCNN) for fruit detection. The proposed system much faster and could be re-trained for different fruits. The bounding box annotation approach was used instead of a pixel-level annotation approach to compute operations faster. Firstly, they were worked on sweet pepper images and after that, the proposed model was tested for

different fruit types. Images that contain different information such as RGB color and near-infrared were used. VGG-16 network model which has 13 convolutional layers was used in the study. Also, faster R-CNN was used to detect fruits. They employed R-CNN and VGG-16 network models together. They tested their proposed system with a small amount of dataset which was collected from commercial farms and grocery stores. Multi-spectral, Kinect 2, and mobile phone cameras were used to collect datasets. For instance, 100 training, 22 testing images were used for sweet pepper. Rockmelon, apple, avocado, mango, and orange fruits also were tried to detect. When fruit size on testing images was different from training images, several false-negative results were detected. They tried to use high-resolution images to handle these false negatives for a small amount of training and testing images. Furthermore, early and late fusion networks were used to improve the DCNN model.

Zhang, Donahue, Girshick, & Darrell (2014) studied fine-grained category detection using part-based R-CNN. Caltech-UCSD bird dataset was used in the study. They tried to detect objects and their parts from images. For instance, birds were detected. Also, the head of birds and bodies of birds were detected and annotated using a bound box. Caffe open-source library was used to experiment with the study. R-CNN was improved to recognize objects and localize their parts. Moreover, SVM was used to extract features from parts. ImageNet pre-trained CNN was fine-tuned for a bird classification task.

Bo, Ren, & Fox (2013) worked on raw RGB-D images and proposed a framework that is hierarchical matching pursuit (HMP). They tried to learning hierarchical feature representations based on unsupervised learning. They tested their approach on five different datasets. Three of them contain RGB-D object images and two of them contain RGB images. Sparse coding was used for the learning phase. The accuracy of the proposed method was increased when using colored images instead of grayscale images. The main point of sparse coding is that, learning a dictionary. This dictionary can be vectors or codes and the K-SVD approach was used for this purpose. Also, HMP was tested for pose estimation of objects.

Eitel, Springenberg, Spinello, Riedmiller, & Burgard (2015) studied multimodal learning using deep CNN for RGB-D images. This type of image contains more information than RGB. Depth information gives the distance between sensor and object. They designed two different CNN for RGB and the depth part. These two CNNs were combined using a late fusion approach. Washington RGB-D object dataset that contains household objects were used to train and test the proposed architecture. CaffeNet framework was used for implementation. CNNs were initialized using ImageNet pre-trained network weights and fine-tunes based on the used dataset. Some preprocessing operations were applied to the dataset. For the depth part, depth information was normalized to between 0 and 255 and 3 channel data were created. Depth data was given as input to CNN in this way.

Ioffe & Szegedy (2015) studied batch normalization. They tried to normalize parameters for each mini-batch in deep networks. ImageNet classification model was used and modified for this purpose. Experiments showed that the same accuracy was caught with fewer training steps according to the unmodified model. Also, when using batch normalization (BN), the dropout approach can be removed from the model because BN encapsulates the dropout model. The training was accelerated with the same or increased accuracy using this approach. Parameters were passed each layer after normalization was applied.

Jia et al. (2014) studied on a framework that Caffe to handle machine learning tasks. Caffe was developed using C++ language. Also, it has bindings for MATLAB and Python. In the first phase of publishing, the framework only supported vision tasks and it was modified and improved to support speech recognition, neuroscience, and astronomy tasks. CPU and GPU computation power can be used in the framework. Training speed can be accelerated using GPU computation power based on the CUDA kernel. Furthermore, they added complete models to the framework for researchers and non-commercial applications. Source code can be reached from GitHub. Four-dimensional array structure, called blobs, was implemented to store data and provide communication between layers.

Krizhevsky (2009) studied Restricted Boltzmann Machines (RBM) and deep belief networks (DBN) which extended version of RBM. Tiny images that were collected from the web were used in the study. RBM is a graphical model that contains two parts that are hidden and visible units. DBN has similar features but it has multiple hidden units. RBM was used to generate weights and these weights were used to initialize the feed-forward network. The generative model was tried to find for classification purposes.

Merler, Galleguillos, & Belongie (2007) worked on a new multimedia database that contains product images and videos from groceries. Data were collected from the web and groceries. Some of the collected images were captured under ideal conditions and this was called vitro. Other parts of collected images were captured under natural conditions and these were called in situ. 120 grocery products were collected under the Grozi-120 dataset. Some algorithms were tested using the Grozi-120 dataset for detection and classification purposes. These algorithms were color histogram matching, SIFT, and boosted Haar-like features. Vitro images were used to train the network and in situ images were used to test the network.

Netzer et al. (2011) worked on detecting and recognizing digits from natural images using feature learning algorithms which were stacked sparse auto-encoders and K-means-based algorithms. They collected street view images that contain house numbers. It is a hard task to detect digits from natural images. Firstly, they tried to locate house numbers from the large image and after that, they tried to recognize characters from detected house numbers using unsupervised feature extraction algorithms that were given above. Furthermore, two more algorithms which were Histogram-OF-Oriented-Gradients (HOG) and binary features were tested using the collected dataset.

Musalem, Olivares, Bradlow, Terwiesch, & Corsten (2010) proposed a method to understands the effects of out-of-stock issues on customer’s behaviors. Out of the stocks' rate is about %8 in the United States and Europe. It is important to analyze OOS to measure financial loss for stores. OOS was captured on fast-moving products

and slow-moving products. It was hard to catch OOS issues for slow-moving products. They tried to analyze issues using data that were taken from store managers. When OOS occurred, 3 options were observed as customer behavior. The first one is that customers waited to purchase the product until available again. Secondly, a customer purchased a substitute product in the same category. Thirdly, the customer did not purchase anything. The third option causes short-term financial effects and the last two options were examined in the study.

2.2 Review of Object Detection Studies for OSA

Moorthy, Behera, Verma, et al. (2015) applied image processing techniques to detect the presence and absence of the product in front of the shelf using MATLAB. Also, they worked on the positioning of products in retail stores. The feature extraction technique was used for object detection and a Speeded Up Robust Features (SURF) algorithm was used for this purpose. Reference images were given to the system and target images were taken from video or camera devices as shelf images. Some pre-process operations were applied to images before comparison of input and target images and extracting features using the SURF algorithm. End of these steps, the product was evaluated for missing or misplaced. Besides, they (Moorthy, Behera, & Verma, 2015) proposed the image processing approach to provide high OSA in retail. They collected images from shelves to use reference images. Reference images and target images were compared using image processing techniques. When comparison showed missing products on the shelf, the system sent messages to the manager or responsible person in the store. A similar template matching approach was used to develop OSA monitoring software in another study (Satapathy et al., 2015). Moreover, some researchers (Rosado, Goncalves, Costa, Ribeiro, & Soares, 2016) worked on high-resolution panoramic images and proposed a supervised learning approach. Panoramic images were created using a wide-angle fisheye camera and an Accelerated-KAZE (AKAZE) feature detector was applied. Labels were detected from the panoramic images of shelves. A Cascade classifier approach was used for this purpose.

Kejriwal, Garg, & Kumar (2015) worked on counting products from shelf images using robot-based equipment. Several cameras were placed on either side of the robot. The robot was moved between shelves and video data were collected from shelves. Several methods were applied to these collected datasets. The first method was used to recognize the product and the second one was used to count the product. A k-d tree was created and the SURF approach was used to recognize the product by using the nearest neighbor search. Two different techniques were used for product count. The first one is the product counting method in which repeating features were counted to understand the number of products on the image by using the SURF method. Secondly, rectangular bounding boxes were drawn around the products and these boxes were counted to understand the number of products by using the Random Sample Consensus (RANSAC) method. Moreover, the grid search approach was applied in the neighborhood of each found product on the image. Some undetected products were tried to find in this way.

Higa & Iwamoto (2018) studied product changes on the shelf. They focused on taken or returned products on the shelf. Videos, captured from a surveillance camera, were used for the study. Low-quality videos, 480x270 pixels, and 1 fps were used to eliminate storage problems but another problem occurred at this time. Moving objects was difficult to track from low-resolution videos. They used background subtraction and convolutional neural network (CNN) approach to detect changes of products on the shelf. CNN was used based on CaffeNet and four classes were determined for CNN. The extended version of the study (Higa & Iwamoto, 2019) proposed analyzing consecutive images when the customers stand in front of the shelf. The Hungarian method was used to analyze consecutive images. Moreover, Canadian Institute for Advanced Research (CIFAR)-10-based network and CaffeNet-based network were used to detect change regions. Also, a heatmap was generated to show the customer's behavior using captured images. Frequently accessed shelves were analyzed with this approach.

Some studies are existing about planogram compliance checking. Planograms are created to standardize placed products on shelves for chain supermarkets. Another aim

of creating a planogram is that, providing the best customer experience for promotions. Liu & Tian (2016) studied automatic planogram checking compliance using recurring patterns. These planograms are created by headquarters and send to store managers. Store managers are responsible for applying planograms to store shelves. In a conventional way, they try to check planogram compliance manually. The proposed automatic compliance checking without using template images. Planogram was taken as XML format and parsed. Some matrix operations were applied to detect recurring patterns. The extended version of the study (Liu, Li, Davis, Ritz, & Tian, 2016) focused on spectral graph matching and speed improvements using a divide-and-conquer approach. Besides, Saran, Hassan, & Maurya (2015) studied visual analysis for planogram compliance. They compared the reference template image to the target image. Their study focused on the presence of products, placements of products, and product count. Hausdorff map approach was used for the presence of products. This presence of products was a group under two classes which were complete and partial cases. Euclidean distance approach was used to identify completely missing cases. A binary distance map approach was used with self Hausdorff map. Shelf rows were identified using Sobel derivatives extractor and Hough line extractor. Sobel derivatives extractor was used for vertical changes and Hough lines extractor was used for horizontal. Finally, texture features and color features were used to count products from images. The color feature was used to eliminate false positives.

Briefly summarized aforementioned works focus on OSA monitoring and planogram checking based on traditional image processing and deep learning techniques using different approaches. Image processing techniques have performance issues for huge datasets. Moreover, for template matching, reference images have to be stored in a database and this is not suitable for real-world applications. On the other hand, labeled images are needed using deep learning techniques for object detection on the shelves. The largest and well-known computer vision datasets do not provide labeled images for store products. This process is time-consuming and quite expensive since it requires human labor costs. This thesis proposes a new method that combines two concepts "semi-supervised learning" and "on-shelf availability" (SOSA) for the first time. The main advantage of the proposed SOSA method is that satisfactory

results can be achieved using a small amount of labeled data and a large amount of unlabeled data for OSA monitoring.

Monitoring on-shelf-availability has very little coverage in the literature, only a few numbers of detailed analyses (Moorthy et al., 2015; Higa & Iwamoto, 2018; Moorthy et al., 2015; Satapathy et al., 2015; Rosado et al., 2016; Kejriwal et al., 2015; Higa & Iwamoto, 2019; Saran et al., 2015) have been performed. Table 2.1 shows the comparison of our study with the previous studies aforementioned. Our approach differs from the existing approaches in many respects and has numerous advantages over the rest as follows:

- First, some of the previous studies (Moorthy et al., 2015; Moorthy et al., 2015; Satapathy et al., 2015; Rosado et al., 2016; Kejriwal et al., 2015; Saran et al., 2015) used traditional techniques such as image processing (IP) to monitor OSA, whereas we used deep learning techniques. In IP-based approaches, a huge amount of reference images has to be stored to match the target image and for every product updating, reference images have to be updated manually. In this context, an important advantage of our method is that it does not require any reference image and therefore it does not need manual updating when products are updated. Moreover, it automatically extracts features from an input image thanks to deep learning.
- Second, the previous deep learning-based studies used different network structures such as CaffeNet-based Network (Moorthy et al., 2015; Kejriwal et al., 2015) and CIFAR-10-based Network (Moorthy et al., 2015; Kejriwal et al., 2015), whereas we designed a novel network architecture that consists of RetinaNet, YOLOv3, and YOLOv4 detectors. Here, the advantage of our approach is that it builds three different models and selects the best one, and hence, satisfactory results can be achieved by the selection of the best model.
- Third, the previous deep learning-based studies used two-stage detectors. On the other hand, in this study, we benefit from one-stage detectors

because of their speed and achieving satisfactory accuracy results for OSA monitoring.

- Four, our study differs from the rest in that we adapted the semi-supervised learning concept, and therefore we benefited from both labeled and unlabeled data. Here, the main advantage is that our method reduces the need for labeling images which is an expensive, tedious, difficult, and time-consuming process since it requires human labor. Satisfactory results can be achieved using a small number of labeled images. Moreover, the proposed method will expand the application field of machine learning in grocery stores since a large amount of OSA data generated in real-life is unlabeled.
- Finally, differently from the previous studies, we introduced an explainable AI concept into OSA. The developed new SOSA XAI software application allows users to manage, understand, and trust the model when monitoring OSA. Besides, we developed an augmented reality application (SOSA AR) to allow users to monitor the last status of the shelf when they are in the shelf section.

Table 2.1 Comparison of the proposed SOSA approach and the previous approaches

Reference	Year	Object Detection			Learning		XAI	Methods
		Traditional Detection	Deep Learning		Supervised Learning	Semi-Supervised Learning		
			Two-Stage	One-Stage				
Moorthy et al.	2015	✔			✔		X	SURF
Moorthy et al.	2015	✔			✔		X	Image Processing
Satapathy et al.	2015	✔			✔		X	Image Processing
Kejriwal et al.	2015	✔			✔		X	k-d Tree, RANSAC
Saran et al.	2015	✔			✔		X	Hausdorff Map, Euclidean Distance, Binary Distance Map
Rosado et al.	2016	✔			✔		X	AKAZE Feature Detector, Cascade Classifier
Higa & Iwamoto	2018		✔		✔		X	CaffeNet-based Network, CIFAR-10-based Network
Higa & Iwamoto	2019		✔		✔		X	CaffeNet-based Network, CIFAR-10-based Network, Hungarian
Proposed Method				✔		✔	✔	RetinaNet, YOLOv3, YOLOv4

CHAPTER THREE
BACKGROUND INFORMATION

3.1 Deep Learning

All features and possible conditions are designed and coded in standard programming methodology. On the other hand, these kinds of developed systems need modifications when new conditions and features appear for the related business. For this reason, it is a time-consuming process to keep alive these systems and sometimes, it is not possible to implement all conditions or features to these systems. At this point, machine learning algorithms come across to handle this situation.

Machine learning algorithms have the ability to describe features, and conditions from previous data to train the machine and for this reason, these algorithms easily adopting condition changes. They are scalable, flexible, and universal. Machine Learning has a lot of sub-fields. One of these sub-fields is deep learning. On the contrary, deep learning has the ability to learn new features from datasets. It has become popular since 2012. Proposed deep learning algorithms started remarkably outperforming previous approaches after 2012. The fundamental part of deep learning models is artificial neural networks that contain multiple layers and each layer takes input data and processing this data to more generic representation and transfer to the following layer (Mureşan & Oltean, 2018). One of deep learning models is Convolutional neural networks (CNN) that are widely used to understand visual data. A sample CNN architecture is shown in Figure 3.1.

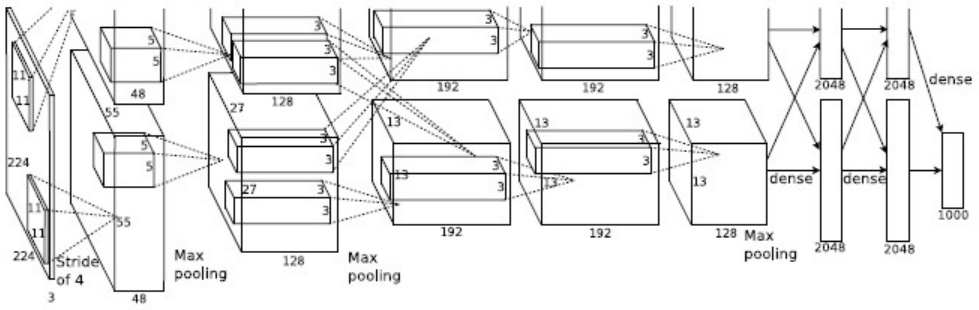


Figure 3.1 A sample CNN architecture (Krizhevsky et al., 2012)

A typical CNN architecture contains convolutional layers, Rectified Linear Unit (ReLU) layers, pooling layers, and fully connected layers. In each layer, matrix operations are processed for visual data. Many different algorithms and frameworks have been developed based on this approach for object detection on images. Also, deep learning algorithms can be grouped under two sub-categories that are one-stage detector and two-stage detector based on object detection.

3.2 Two-stage Detectors

Two-stage detectors such as Regions with Convolutional Neural Networks (R-CNN) (Girshick, Donahue, Darrell, & Malik, 2014), Faster R-CNN (Ren, He, Girshick, & Sun, 2017), Feature Pyramid Network (FPN) (Lin et al., 2017) classify objects in two phases. In the first phase, candidate regions are generated from the input image and after that, classes and bounding boxes are created using these regions.

3.2.1 R-CNN

Girshick, Donahue, Darrell, & Malik (2014) proposed a new algorithm, and the proposed approach has better performance values compared to previous approaches. For this comparison experiment, the well-known PASCAL VOC 2010 image dataset was used. R-CNN consists of three sections that are region proposals, feature extraction, and classification. R-CNN architecture is shown in Figure 3.2.

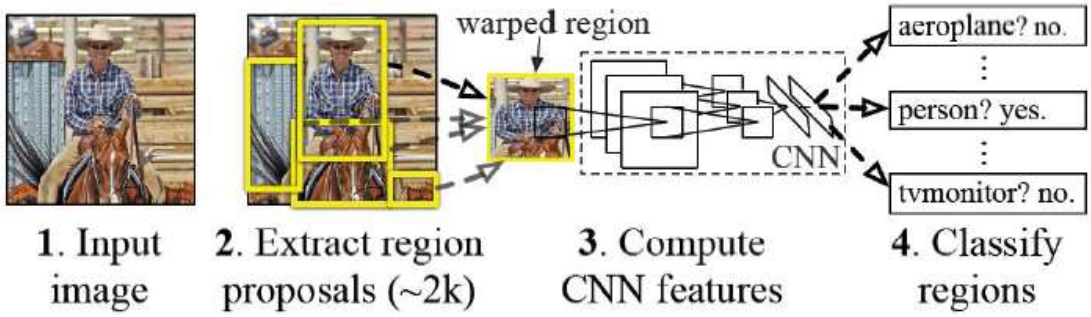


Figure 3.2 R-CNN architecture (Girshick et al., 2014)

The main flow of the algorithm is that the input image is given and region proposals are generated according to color and texture similarities from the image and the selective search algorithm is used for this purpose. In the second phase, features are extracted for each proposal region using CNN. In the final phase of the algorithm, regions are classified using the SVM approach.

3.2.2 Faster R-CNN

Ren, He, Girshick, & Sun (2017) proposed a new algorithm, and the proposed approach solved the speed problem in the region extraction phase. In this phase, they implemented the region proposal network to solve the problem. Previous algorithms used the selective search approach instead of this new approach. Faster R-CNN architecture is shown in Figure 3.3.

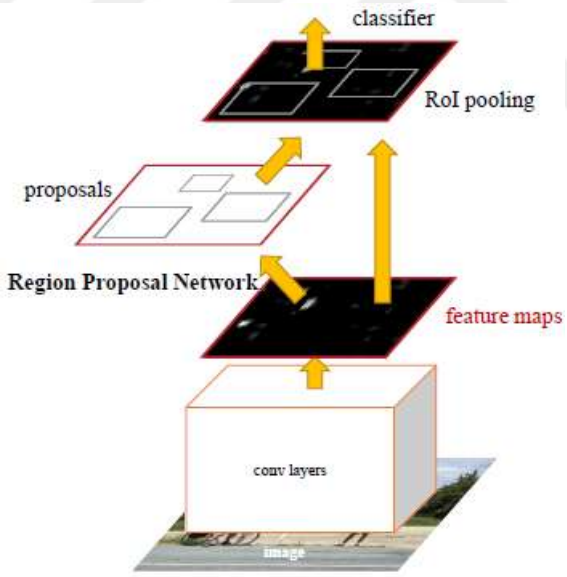


Figure 3.3 Faster R-CNN architecture (Ren et al., 2017)

An input is given to the region proposal network and the network output is generated object proposals as rectangular shapes and each region proposal has a score value. Finally, all these outputs are processed in a fully convolutional layer. Feature maps are created and these feature maps are connected to the classifier layer. Figure 3.4 shows the detailed architecture of the region proposal network and sample detection results.

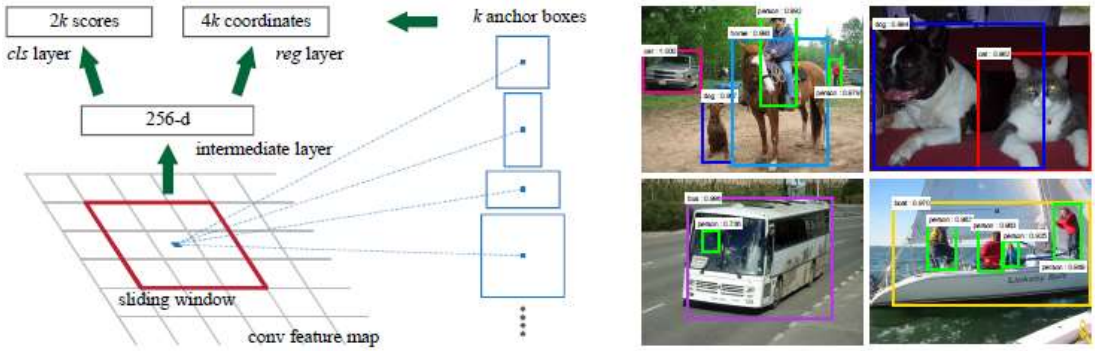


Figure 3.4 (Left) The detailed architecture of the region proposal network; (Right) Sample detection results (Ren et al., 2017)

3.2.3 FPN

Lin et al. (2017) proposed a new approach that was the feature pyramid network. The aim of the proposed approach is that creating a feature pyramid and the top of the pyramid has high-resolution strong features and the bottom of the pyramid has weak features. Architecture of the FPN approach is shown in Figure 3.5.

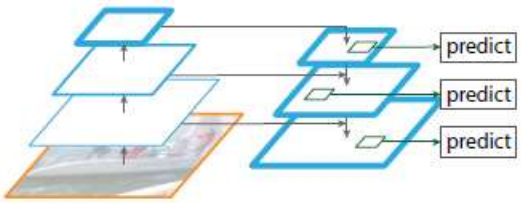


Figure 3.5 The FPN architecture (Lin et al., 2017)

In this approach, an input image is given and different scale feature maps are generated from the given image. Better accuracy results were taken using these generated features. COCO dataset was used for this experiment. This was proposed as a generic solution and it can be implemented for different convolutional neural networks such as RPN and Fast R-CNN.

3.3 One-stage Detectors

One-stage detectors such as RetinaNet (Lin, Goyal, Girshick, He, & Dollar, 2018), YOLOv3 (Redmon & Farhadi, 2018), and YOLOv4 (Bochkovskiy, Wang, & Liao, 2020) classify objects at the one-stage on the contrary of two-stage detectors. In this approach, the candidate regions part is skipped. Bounding boxes and classification results are extracted from the input image.

3.3.1 RetinaNet

Lin, Goyal, Girshick, He, & Dollar (2018) proposed new one-stage detector network architecture together with a novel loss function to handle class imbalance issues. This approach allows dynamic scaling for cross-entropy loss. In the experimental study, high accuracy results were taken based on bounding box detection and outperformed the previous approaches included in the two-stage detectors' category. For the experimental studies, the COCO dataset was used. RetinaNet architecture is shown in Figure 3.6.

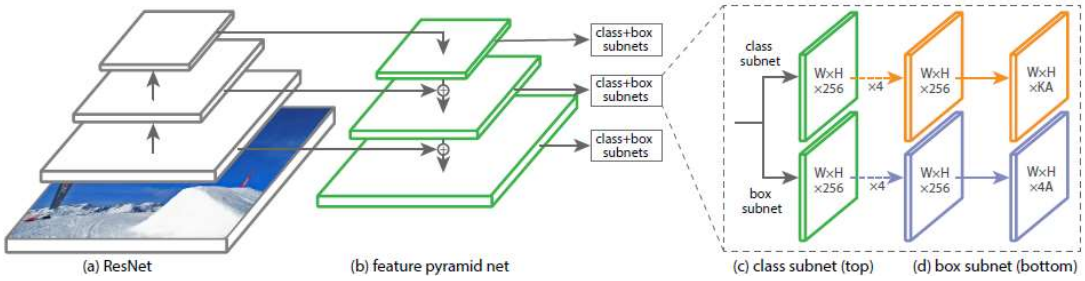


Figure 3.6 RetinaNet architecture (Lin et al., 2018)

It uses ResNet architecture as a backbone and the feature pyramid network approach is used for feature extraction. Also, two subnetworks are used. One of them is used to classify anchor boxes and the other one is used to compare predicted anchor boxes to ground-truth boxes.

3.3.2 YOLOv3

YOLO is a real-time object detection framework and it uses a convolutional based neural network as the backbone. On the contrary of previous approaches, YOLO resizes and divides input images using the S x S grid and compares predicted bounding boxes to ground-truth bounding boxes based on intersection over union instead of the region-based approaches. It predicts classes according to predicted box confidence values (Redmon, Divvala, Girshick, & Farhadi, 2016). YOLO model is shown in Figure 3.7.

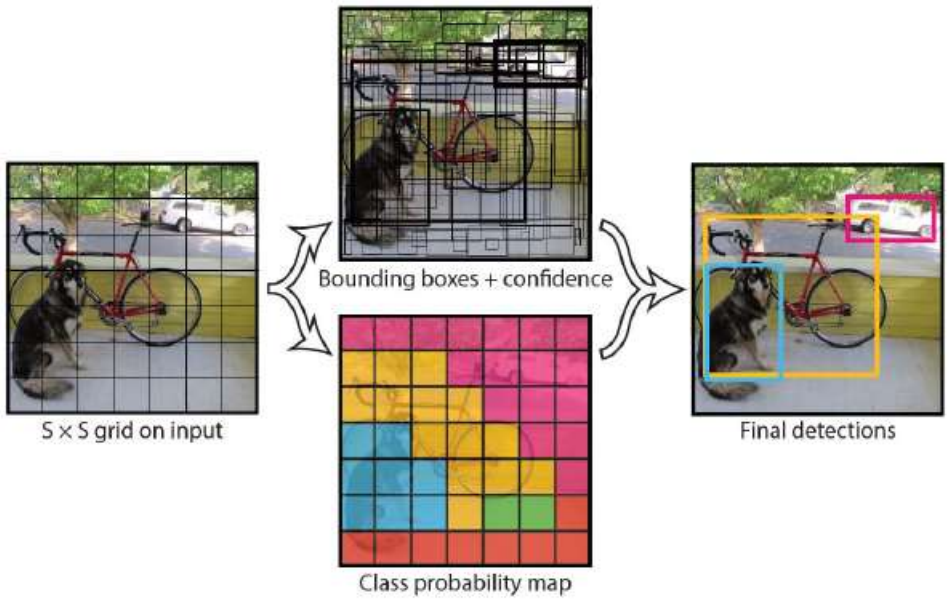


Figure 3.7 YOLO model (Redmon et al., 2016)

Redmon & Farhadi (2018) improved the YOLO framework in this study. A new convolutional neural network called Darknet-53 was proposed as the backbone that has 53 convolutional layers. The previous version of the YOLO is used Darknet-19 as the backbone. Besides, some shortcuts were added to the proposed network. The network was tested using the COCO dataset and results outperformed previous approaches for both inference time and average precision value.

3.3.3 YOLOv4

Bochkovskiy, Wang, & Liao (2020) proposed an improved version of previous YOLO frameworks. Significant features were added to the proposed framework called YOLOv4. These features are weighted-residual-connections (WRC), self-adversarial-training (SAT), cross-stage-partial-connections (CSP), cross mini-batch normalization (CmBN), mish-activation and Mosaic data augmentation. YOLOv4 architecture is shown in Figure 3.8.

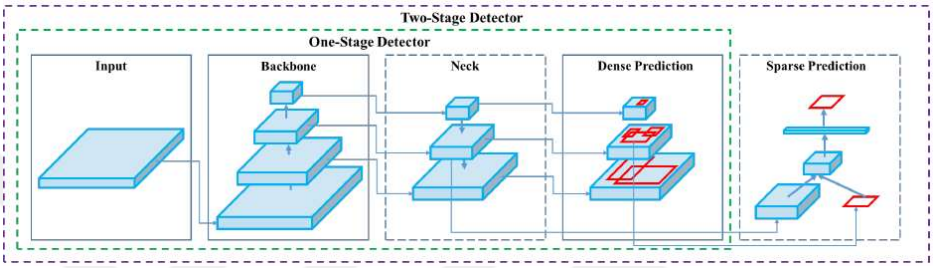


Figure 3.8 YOLOv4 architecture (Bochkovskiy et al., 2020)

The proposed model can be trained on a single GPU and high accuracy results can be taken. Cross Stage Partial Darknet-53 (CSPDarknet53) architecture was proposed as the backbone. The path-aggregation network (PANet) was implemented for detection steps. Spatial Pyramid Pooling (SPP) blocks were added. Also, the Mosaic data augmentation approach that combines four images was added to improve the accuracy in the training phase. Sample images after the Mosaic augmentation step are shown in Figure 3.9.

Figure 3.9 Sample images after the Mosaic augmentation step (Bochkovskiy et al., 2020)

CHAPTER FOUR
MATERIALS AND METHODS

4.1 Deep Learning for Object Detection

One of the main tasks of computer vision is object detection. It deals with detecting objects along with their locations and classes (such as cars, fruits, food products) in images. Object detection methods can be grouped into two categories: traditional image processing methods and deep learning-based detection methods. Recently, deep learning-based object detection methods have become popular because of taken outstanding results. In addition, deep learning-based object detection methods can be divided into two categories as two-stage detectors such as Regions with Convolutional Neural Networks (R-CNN) (Girshick et al., 2014), Faster R-CNN (Ren et al., 2017), Feature Pyramid Network (FPN) (Lin et al., 2017) and one-stage detectors such as RetinaNet (Lin et al., 2018), YOLOv3 (Redmon & Farhadi, 2018), and YOLOv4 (Bochkovskiy et al., 2020). In the general structure of two-stage detectors, feature extraction is applied to the input image and generated proposed regions using different methods in the first stage. From these proposed regions, the locations (bounding boxes) and classes of the objects are determined in the second stage. In contrast to two-stage detectors, the region proposal stage is skipped and directly learned class probabilities and bounding box locations from the input image like a simple regression problem. In this study, we have decided to work on one-stage detectors because of their speed and achieving satisfactory accuracy results for OSA monitoring.

The history of one-stage detectors is shown in Figure 1. Since 2012, CNN (Krizhevsky et al., 2012) has been used to be able to learn from features of images robustly. Before this year, traditional detection methods such as image processing were used for object detection tasks. After this year, computer vision techniques have improved very rapidly and novel deep learning-based detection methods have been proposed. The existing detector methods can be grouped under two categories: methods with one-stage detector and methods with two-stage detectors. The first one-stage detector method was proposed in 2013, and since then, at least one novel approach has been proposed each year. Each approach solved the object detection

problem from a different perspective to increase accuracy, and each one achieved higher accuracy results from previous ones. For this reason, three of the latest published one-stage detectors (RetinaNet, YOLOv3, YOLOv4) have been selected for this study.

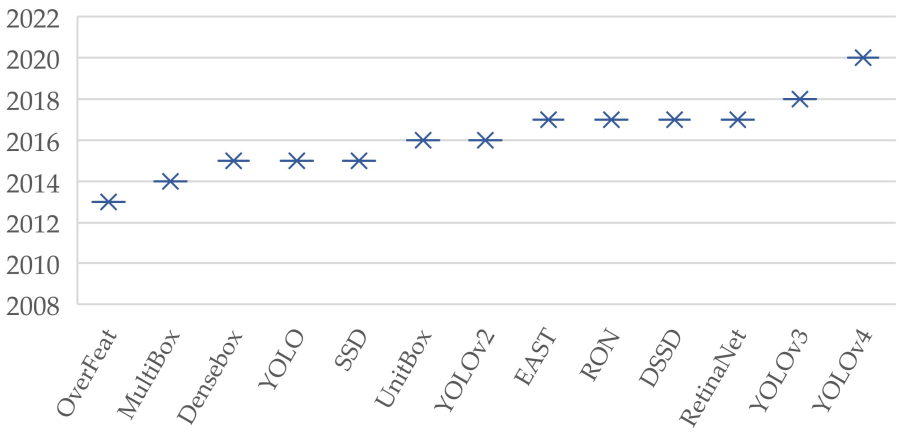


Figure 4.1 History of one-stage detectors

Lin et al. (2018) proposed RetinaNet one- stage object detection model. It uses a new focal loss function to handle class imbalance issues more effectively according to alternative previous approaches. Their model has one backbone network and two subnetworks. Feature map is computed in the backbone network and the output of this network is used as input for two subnetworks. Bounding boxes and classifications are found separately in each subnetwork. FPN is used as a backbone network for RetinaNet. Their proposed architecture improved the performance of the model compared to standard top-down CNN architecture.

YOLOv3 is a one-stage detector that combines FPN, CNN, and non-maximum suppression algorithm (Redmon & Farhadi, 2018). CNN is used for the feature extraction process. In addition, it has been integrated with the Darknet-19 and Residual Neural Network (ResNet) network structures for feature extraction. On the other hand, a shortcut connection is added to the model. Fifty-three convolution layers are existing with dropout and batch normalization operations in the feature extraction network. The network is used as a backbone network and named Darknet-53. YOLO works on the entire image during the train and test processes.

YOLOv4 is an improved version of previous YOLO models. It combines the YOLOv3 head, and path-aggregation network (PANet) for detection steps (Bochkovskiy et al., 2020). PANet is used instead of FPN in the model. The novel backbone, called Cross Stage Partial Darknet-53 (CSPDarknet-53), is used. Some blocks are added such as Spatial Pyramid Pooling (SPP) to increase the receptive fields on the backbone. Besides, YOLOv4 provides data augmentation to expand the training dataset to improve the accuracy of the network without extra inference time. The Mosaic data augmentation method that combines four images in the training phase was proposed.

4.2 Proposed Approach: Semi-Supervised Learning on OSA (SOSA)

This thesis proposes a new approach: Semi-Supervised Learning on OSA (SOSA). The proposed method combines two concepts "semi-supervised learning" and "on-shelf availability" for the first time to decrease OOS by automatically detecting and classifying products using shelf images. For this purpose, SOSA builds a classification model that allows the detection of “Product”, “Empty Shelf”, and “Almost Empty Shelf” classes by using both labeled and unlabeled image data. The aim of the SOSA approach is section-based detection of empty and almost empty shelves based on the semi-supervised learning principle.

4.2.1 The General Structure of the Proposed SOSA Method

Figure 4.2 shows the general structure of the proposed SOSA approach. SOSA consists of the following main steps. The first step is to train three one-stage detectors on the existing labeled image data. After that, the best-trained model is selected using the evaluation metrics such as Mean Average Precision (mAP), F1-score, and Recall measures. In the next step, the unlabeled image data is labeled by the constructed classifier via their predictions, which is commonly referred to as pseudo-labeled data. Lastly, the final classifier is built by using both originally labeled and pseudo-labeled

image data. Hence, SOSA allows unlabeled image data to be introduced to the training process in an efficient manner.

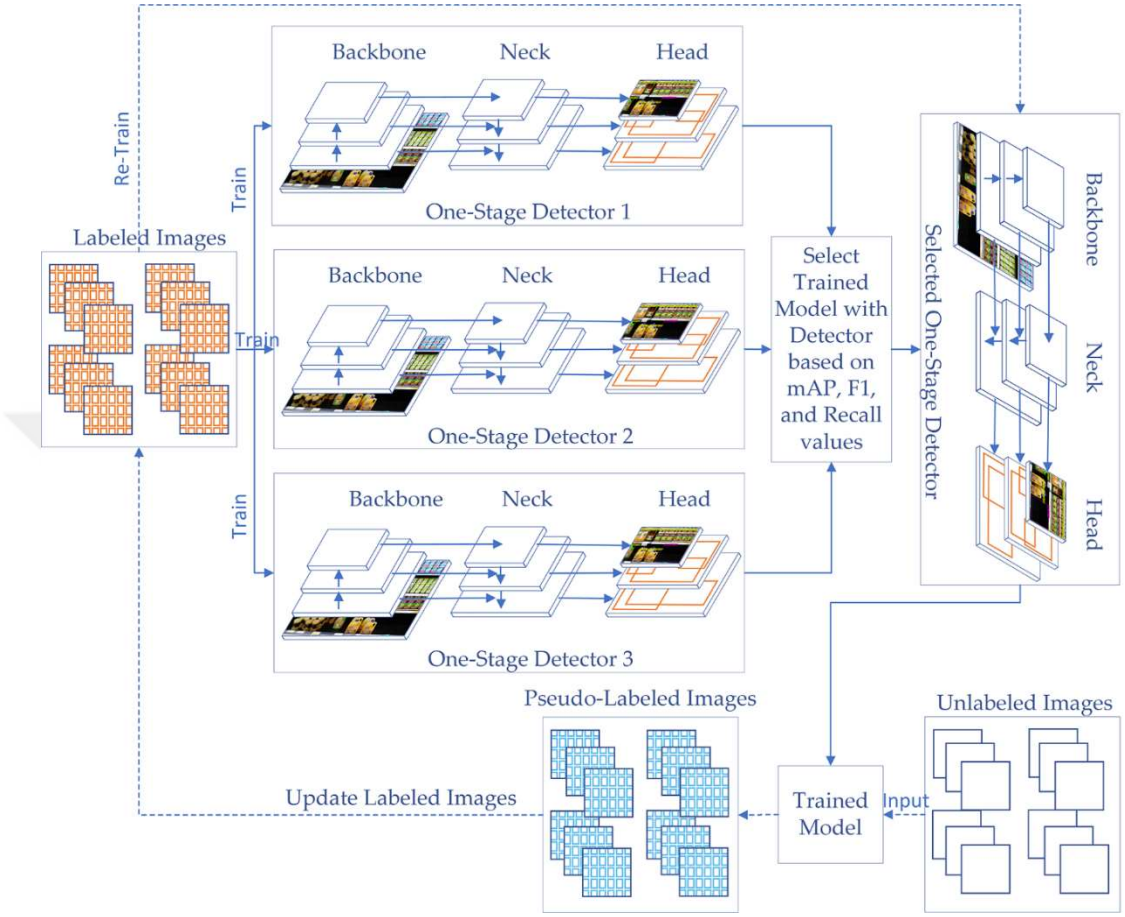


Figure 4.2 The general structure of the proposed SOSA approach

In the preprocessing phase of the SOSA method, the images, which are taken from different perspectives in front of the shelves, are labeled to use for the first training. Images are labeled according to the product type(s) it includes. Besides, empty and almost empty shelves are labeled on the images. If one or two of the related products remained on the shelf, these products' areas are labeled as "Almost Empty".

4.2.2 The Formal Definition of the Proposed SOSA Method

Assume that labeled dataset $D = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$ has n images with labeled products. Each component (x, y) is composed of d-dimensional vector (x) from a given input space X, such that $x \in X$, and the output variable (y), where $y \in Y = \{c_1, c_2, \ldots, c_m\}$ has m class labels. Unlabeled dataset $U = \{x_{n+1}, x_{n+2}, \ldots, x_{n+s}\}$ has s unlabeled images. We are especially interested in OSA images where labeling the images is difficult and expensive since it requires human labor. The SOSA method considers both D and U to find a decision function $f : X \mapsto Y$ that can correctly predict the class labels $\hat{y}$ of a given unseen input sample image SI. Z refers to one-stage detectors and $Z = \{z_1, z_2, \ldots, z_k\}$ has k detectors. In this study, three one-stage detectors are used, where z_1 refers to RetinaNet, z_2 refers to YOLOv3, and z_3 refers to YOLOv4, and so k is set to 3.

Definition 1. *Semi-supervised learning on on-shelf-availability* (SOSA) refers to a machine learning approach that builds a model to correctly detect "Product", "Empty Shelf" and "Almost Empty Shelf" regions from an input image by using both labeled and unlabeled image data, which are taken from different perspectives in front of the shelves.

Algorithm 4.1 gives the general framework of the proposed SOSA approach. The algorithm consists of five steps. In the first step, labeled dataset D is split as D_{Train}, and D_{Test} based on the given percentage. In the first loop, a model z_c is trained for each one-stage detector. Trained models are added to Z. In the second step, the constructed models are tested using labeled test dataset D_{Test} and obtained prediction results are compared to select the best one-stage detector that has maximum success rates. The selected-one stage detector is assigned to SD. In the third step, a query instance $x_i \in U$ is submitted to the selected detector SD, and its estimation $\hat{y}$ is assigned to x_i as a pseudo-label. This process is repeated for each instance in the unlabeled dataset U to generate all pseudo labels. At the end of this labeling operation, labeled dataset D is augmented with pseudo-labeled images. In the next step, the classifier is re-trained by using the new dataset D which contains both labeled and pseudo-labeled images.

Finally, a sample image *SI* is given to the trained model *TM* to be classified, and the predicted classes $\hat{y}$ are obtained from the algorithm.

Algorithm 4.1- SOSA: Semi-Supervised Learning on OSA.

Inputs: D: Labeled dataset $D = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$ with n instances
U: Unlabeled dataset $U = \{x_{n+1}, x_{n+2}, \ldots, x_{n+s}\}$ with s instances
Z: One-stage detectors $Z = \{z_1, z_2, \ldots, z_k\}$ with k detectors
Y: Class labels $Y = \{c_1, c_2, \ldots, c_m\}$ with m classes
SI: Sample image
Outputs: *TM*: Trained model
ŷ: Predicted class labels for the products included in the sample image
Begin:
D_{Train} = Split(D, n * *percentage*)
D_{Test} = Split(D, (n – (n * *percentage*)))
// Step 1 – Training with labeled data
for c = 1 **to** k **do**
foreach epoch
foreach (x_i, y_i) **in** D_{Train}
z_c = Train(x_i, y_i)
end foreach
end foreach
$Z = Z \cup z_c$
end for
// Step 2 – Testing one-stage detectors and selecting the best one
for c = 1 **to** k **do**
foreach (x_i, y_i) **in** D_{Test}
Prediction = $z_c(x_i)$
$PredictionResult_c = PredictionResult_c \cup Prediction$
end foreach
end for
SD = MAX($PredictionResult_c$) // SD: Selected detector
// Step 3 – Labeling unlabeled image data and generating pseudo-labels
foreach x_i **in** U
$\hat{y}$ = SD(x_i)
D.Add(x_i, $\hat{y}$)
end foreach
// Step 4 – Re-training the model with pseudo-labeled data
TM = Train(D)
// Step 5 – Classifying a sample image
$\hat{y}$ = TM(*SI*)
End Algorithm

After the preprocessing phase, the algorithm splits the labeled data into training and test sets according to a given ratio. For each one-stage detector, a model is built by using the training set. At the end of the training phase, three trained models are tested using the test set. As test results, the Average Precision (AP), Mean Average Precision (mAP), F1-score, and Recall values are calculated based on Intersection Over Union (IoU) threshold value. IoU computes the intersection over the union of the given

bounding box and the predicted bounding box. The formulas are given in Equation (4.1) to (4.5).

$$Precision = \frac{True\ Positive\ (TP)}{True\ Positive\ (TP) + False\ Positive\ (FP)} \tag{4.1}$$

$$Recall = \frac{True\ Positive\ (TP)}{True\ Positive\ (TP) + False\ Negative\ (FN)} \tag{4.2}$$

$$F1Score = 2\ .\ \frac{Precision\ .\ Recall}{Precision + Recall} \tag{4.3}$$

$$AP = \frac{1}{N}.\ \sum_{i=0}^{N} Precision_i \tag{4.4}$$

$$mAP = \frac{1}{|T|}.\ \sum_{t\epsilon T} AP_t \tag{4.5}$$

where $|T|$ is total of all APs calculated for each class and N is the number of instances in the test set.

After the testing phase, the obtained test results are compared with each other and the detector that has the highest accuracy is selected as the final representative model. In the following phase, semi-supervised learning is performed by labeling the unlabeled data using the trained model of the selected detector. After the labeling operation is completed, the pseudo-labeled data is combined with the labeled data, and then the training process is restarted on the selected detector using the extended dataset. At the end of the re-training operation, the newly trained model is saved for further use for the detection of empty and almost shelves based on sections. To understand which section has empty or almost empty shelves, the Relative Frequency (RF) formula is used as given in Equation (4.6).

$$RF = \frac{Frequency\ of\ One\ Product\ Class}{Total\ Frequencies\ of\ All\ Products'\ Classes} \tag{4.6}$$

Each product is labeled at the end of the detection phase on the shelf image and these labels denote the section of products. For each section, RF is calculated and the highest RF value gives the final section info. For instance, if the highest RF value is

obtained for the breakfast products, the section is recognized as a breakfast product. Besides, if three empty and two almost empty shelves are detected on the image, the SOSA algorithm gives the following information: “3 empty and 2 almost empty shelves are existing on the breakfast section.”

4.2.3 The Advantages of the Proposed SOSA Method

The proposed method (SOSA) has a number of advantages that can be summarized as follows:

- The traditional OSA applications are limited to using only labeled image data to build a model. However, labeling shelf images are a time-consuming, tedious, expensive, and difficult job because of existing so many products on a one-shelf image, and for this reason, so many human laborers are needed. This is especially true for the OSA applications that include learning from a large number of class labels and distinguishing similar classes. The main advantage of the SOSA method is that it solves OSA problems using a small number of labeled shelf images. The existing labeled dataset is extended by using unlabeled data with automatically assigned labels, and hence, high accuracy results are taken with the SOSA approach in an efficient way.
- Another advantage is that it includes three different one-stage detector models and the model with the highest accuracy is selected at the beginning of the semi-supervised learning. Hence, satisfactory results can be achieved by the selection of the best model.
- The SOSA approach uses three different deep learning techniques (RetinaNet, YOLOv3, and YOLOv4) without any modification or development of the methods. Therefore, SOSA has advantages in terms of easy implementation. It is possible to implement it in Python by using open-source codes available in the related machine learning libraries.
- The main idea behind the SOSA method is to take advantage of a huge amount of unlabeled image data when building a classifier. In addition to labeled data, the SOSA method also exploits unlabeled data to improve classification performance. Thanks to the SOSA method, the unlabeled data

instances provide additional knowledge, and they can be successfully used to improve the generalization ability of the learning system.

- Another advantage is that the SOSA method can be applied to any OSA image data without any prior information about the given dataset. It doesn't make any specific assumptions for the given data.
- Since a large amount of OSA data generated in real-life is unlabeled, the SOSA method will expand the application field of machine learning in grocery stores.

4.3 Explainable AI for SOSA

*Explainable artificial intelligence (*XAI) is a growing research field in recent years. The aim of XAI is to make machine learning and AI applications more understandable to users who are not experts in these fields. From rule-based systems to deep learning systems, the transparency of systems is decreasing. Especially, it is hard to understand and interpret the outputs of deep learning applications by users. Therefore, users see this kind of AI application as a black box. They think that inputs are given to these applications; afterward, something happens magically inside the box, and outputs are generated by AI applications (Emmert-Streib, Yli-Harja, & Dehmer, 2020). On the other hand, AI-based applications are increasingly being adopted by different sectors, including retail. Therefore, business stakeholders and users of AI systems should be able to understand their systems to trust outputs and manage these applications to their needs without getting help from AI experts or engineers (Confalonieri, Coba, Wagner, & Besold, 2020).

There are different concepts of XAI to provide transparency for AI models in the literature. This transparency can be provided for different parts of the model based on requirements. One of them is post-hoc explainability. The purpose of post-hoc is to explain decisions of AI models using different approaches such as text explanation, visualization, explanation by example (Barredo Arrieta et al., 2020). The results of the model can be explained using text definitions, graphics, and images for users in this

way. Besides, the dataset can be interpreted during the training phase and the dataset can be extended for better accuracy without the need for AI experts.

In this study, we present the first demonstration of XAI on OSA using post-hoc techniques, and therefore, users can understand and trust the constructed OSA AI model in this way. Moreover, they can interpret the detection results of the model and enhance the dataset in the training phase to adapt the application for changes. In this study, for the OSA XAI demonstration, we used the following post-hoc techniques: text explanations, visualizations, and explanations by example.

In this study, a new software application, called SOSA XAI, was developed to provide understandability for the users. SOSA XAI consists of four main screens. These are “Train”, “Test”, “Monitoring”, and “Metrics” screens. The main screen is shown in Figure 4.3. Our proposed SOSA method can be managed using the application. After the first initialization, users can re-train the model using the training screen as shown in Figure 4.4. In the training screen, the creation dates of the models and the obtained results (mAP, F1-score, and Recall values) are given in an easily understandable format. Furthermore, for each model, the status of the model is shown based on accuracy values using the visualization technique. If one of these three accuracy results under 80% (our threshold value for this study), color is changed for the related accuracy metric. Moreover, during the training process, training progress steps are given in a more understandable way using the text explanation technique, and at the end of the training operation, the meaning of accuracy values are interpreted by the application. The last trained model has activated automatically. On the other hand, users can activate one of the previous models from the list.

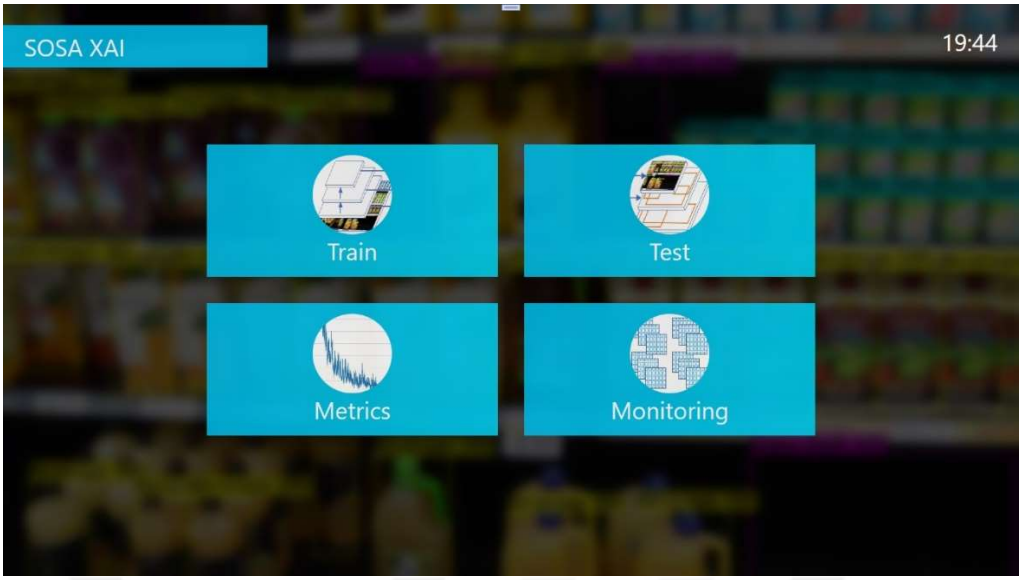


Figure 4.3 The main screen of SOSA XAI

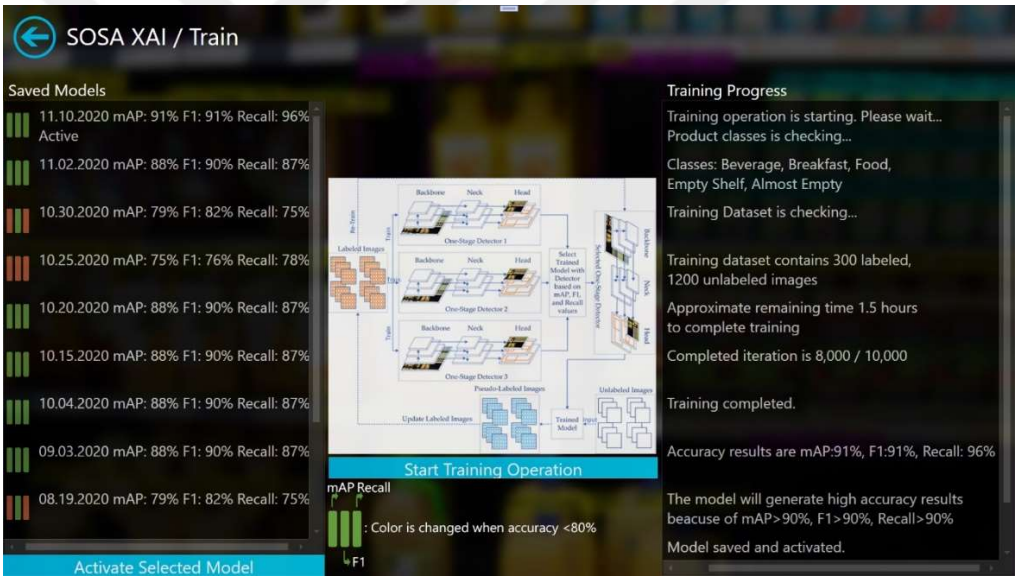


Figure 4.4 Train screen of SOSA XAI

Thanks to our SOSA XAI application, the active trained model can be tested using a test screen. Test images are listed on the screen and also new test images can be uploaded. Accuracy values and creation date of the trained model is shown in a straightforward manner. For selected or uploaded test images, detection operation can be started and results are shown both visually and textually. In the textual explanation section, the decision of the model is given an understandable way and explained how this decision is made. Figure 4.5 shows a screenshot of the object detection results.

Besides, when some of the classes are detected with low accuracy, the application gives suggestions to extend the existing dataset to increase accuracy. For instance, when empty-shelf accuracy is less than 80%, the system suggests adding more images that contain more empty shelves from different perspectives and re-training the model. In the metrics screen of the software, the system shows sample images containing empty shelves from different perspectives, and users can expand the training set by adding new images among these samples. Moreover, the monitoring screen has also similar functionalities, but here, the user can design the test set, instead of the training set. From the monitoring screen, real-time results can be tracked the same as the test screen.

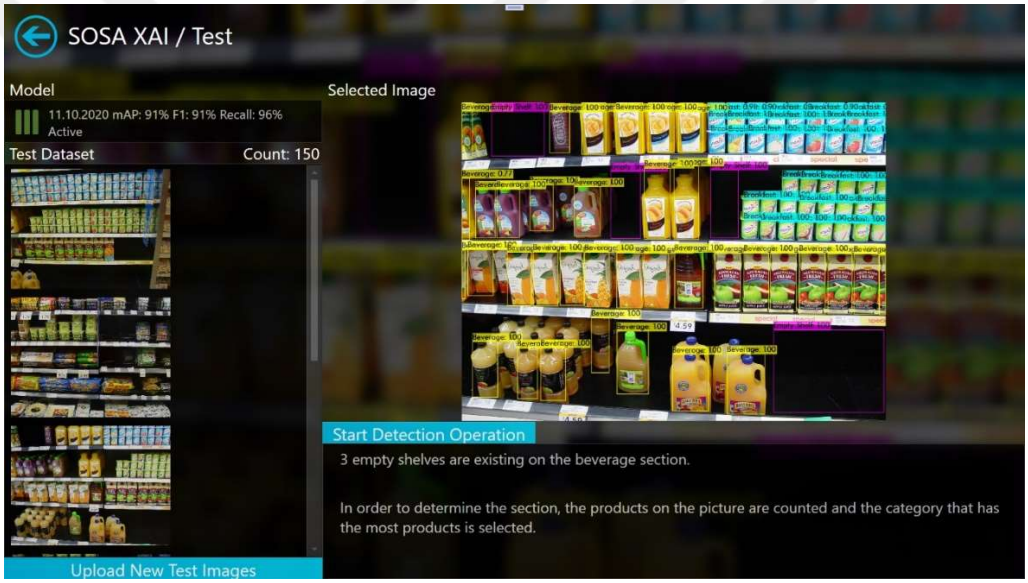


Figure 4.5 Test screen of SOSA XAI

Finally, from the metrics screen (Figure 4.6), the training accuracy graph for the active trained model, and dataset distribution based on classes can be seen easily. In addition, the dataset can be extended using the "Extend Dataset" part with unlabeled images. Sample images are shown to give an idea of how to create a new dataset and suggestions are given as text for this purpose. After the upload operation is completed, the training process can be started from the training screen for the extended dataset. Text explanation, visualization, and explanation by example post-hoc techniques are used in this screen.

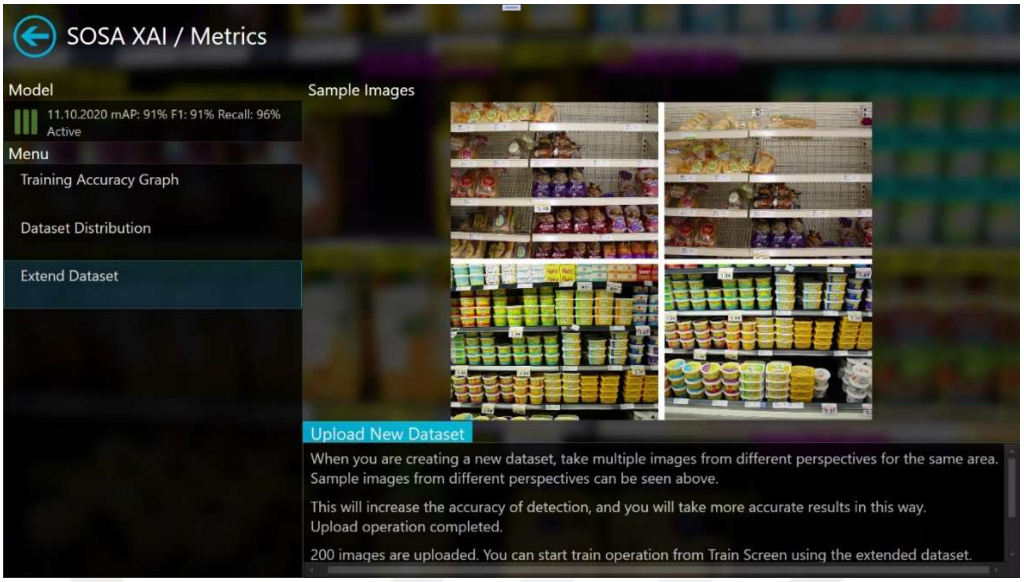


Figure 4.6 Metrics screen of SOSA XAI

In a real-time application, shelf images can be taken approximately every hour during peak times and every three hours during non-peak times to be analyzed by the OSA system. Since this period is parametric in the developed system, it can be easily modified. In order to protect the privacy of customers, when an image is taken, an object detection approach (Kajabad & Ivanov, 2019) is firstly applied to check the presence of people on the image. If people are detected, this image is not processed by the system, and a new image is taken after a period of time until the image does not contain a person (i.e., shopper or employee). After this step, the trained model is used to classify the new image, and the image is stored in the system. When empty or almost-empty shelves are detected, a notification is sent to the responsible store clerk to check the related area. When a notification is received, a responsible clerk can put new products on the shelf to prevent customer and profit loss.

SOSA XAI gives an opportunity to understand, trust, and manage AI applications to increase OSA for users who are not AI experts and engineers in grocery stores. Results can be interpreted easily and the dataset can be extended with unlabeled images for requirements changes. Furthermore, the results of the previous models can be compared to the current one and the desired model can be selected and activated. It combines our proposed SOSA approach and XAI.

4.4 Augmented Reality Application for SOSA

The augmented reality (AR) field has become popular with the advancement of technology in recent years. In AR applications, Virtual objects, texts, 2D, and 3D images are placed in a real-world environment. This technology builds a connection between the virtual world and the real world (Azuma, 1997). This technology can be implemented using different devices such as computers, tablets, and smartphones. In this study, a mobile application, called SOSA AR, was developed using AR technology. When developing the application, the JAVA programming language and ARCore framework were used. The application is integrated to SOSA, SOSA XAI, and the database via RESTful API. The system architecture is shown in Figure 4.7.

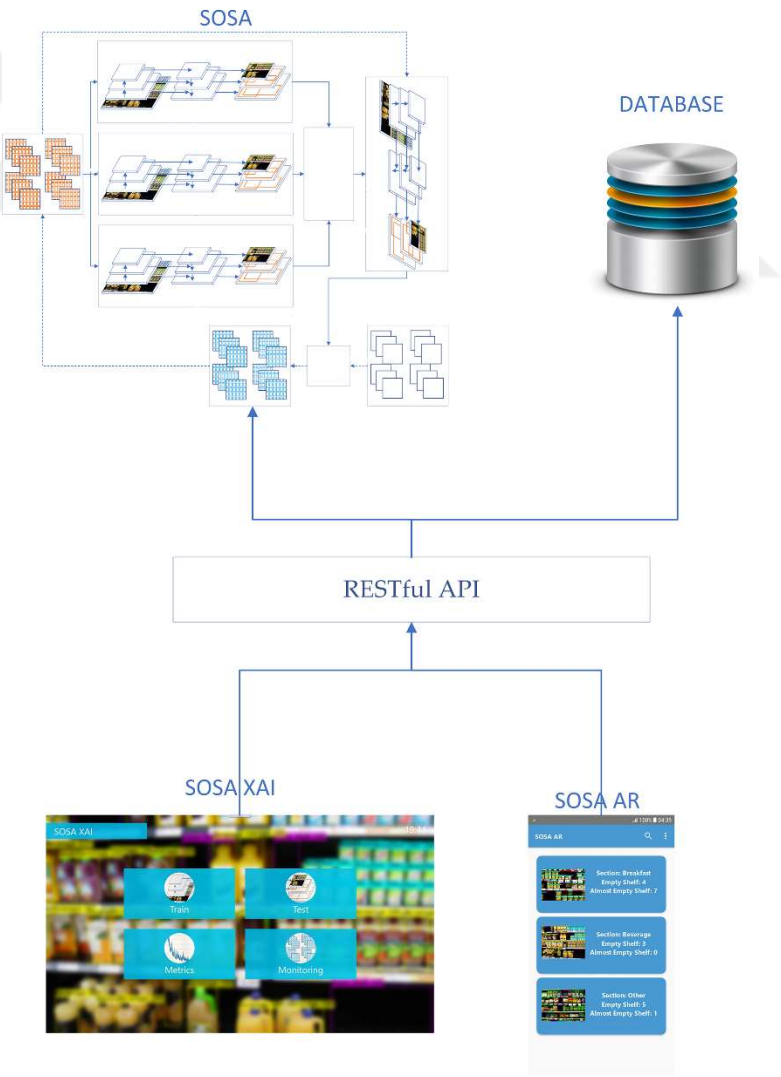


Figure 4.7 Architecture of SOSA AR

SOSA AR provides notification monitoring and last audit monitoring features for responsible clerks who manage shelves. Notifications that are taken from SOSA is listed on the notification screen. Responsible clerks can check notifications from the screen and take action about the notification. The notification screen is shown in Figure 4.8. In the notification detail, the number of empty and almost empty shelf are given for each section. Also, visual detection results are given and users can see the location of empty and almost shelves.

Figure 4.8 Notification screen of SOSA AR

In addition, the last audit time with date information can be seen for each section. For this purpose, users hold the smartphone towards a shelf and SOSA AR recognizes the section field automatically using augmented reality technology and returns the last audit info on the instant monitoring screen. The instant monitoring screen is shown in Figure 4.9.

Figure 4.9 Instant monitoring screen of SOSA AR

The augmented images approach was used to recognize sections from shelf images. For this purpose, an augmented image database was created. Thirty images, taken from different perspectives and shelf parts, were added to the augmented image database for each section. These images are used as a reference image. When the user holds the smartphone towards a shelf, the taken image from the camera is compared to these reference images using simple computer vision feature extraction techniques. If the taken image from the camera is matched to any reference image, the ARCore framework returns class information of the reference image, and the application process the algorithm for the following operations.

CHAPTER FIVE
EXPERIMENTAL STUDIES AND RESULTS

5.1 Dataset Description

In the experiments, the proposed approach SOSA was tested on a real-world dataset. A labeled dataset is needed for the test operation but the largest and well-known computer vision datasets do not provide annotation for grocery store products. In the literature, few datasets have been collected from grocery stores. On the other hand, all of these datasets have unlabeled images for the store products, and these datasets cannot be used directly without labeling some of them.

We used the WebMarket dataset ("WebMarket is an image database built for Computer Vision Research," n.d.) in this study. The dataset contains 3153 unlabeled images that were taken from in front of shelves using three digital cameras that were standing from about 1 meter away from shelves. The images were taken without any special illumination changes and without any viewpoint constraint but most of the images are frontal views. The images in the dataset were collected from 18 shelves in a retail store, each of length 30 meters and each of which approximately has six levels. Each image generally covers an area of about 2 meters in width and 1.5 meters in height on shelves, including all the items within three or four shelf levels in range. The dataset contains images of 100 different items. The dataset also includes fine-grained product categories having minor variations in packages (Santra & Mukherjee, 2019). Since the model is trained on such data, it has the capability of dealing with packages. The format of each image is Jpeg and its resolution is either 2592 x 1944 or 2272 x 1704. High resolution holds sufficient information for each object appearing in the image, and thus, the trained model can deal with packages of multiple items and damaged packages, at least with lower accuracy. Until now, the dataset was used for object retrieval studies (Y. Zhang, Wang, Hartley, & Li, 2007, 2009). Namely, our study is the first study that uses the WebMarket dataset for monitoring on-shelf-availability.

5.2 Experimental Studies

In order to label the images, LabelImg ("LabelImg," n.d.) open-source tool was used and 300 images were labeled based on five classes. Three of them are for product categories that are "Beverage Products", "Breakfast Products", and "Food Products", and two of them are for shelves' areas that are "Empty Shelf" and "Almost Empty Shelf". "Beverage Products" and "Breakfast Products" were selected from fast-moving consumer goods (FMCG) which are nondurable products, consumed at a fast pace. Total 13835 products grouped under three categories, and 818 shelves' areas grouped under two categories. The labeled dataset was divided into a training set (90%) and a test set (10%). The distribution of labeled products and shelves' areas are shown in Figure 5.1 and Figure 5.2.

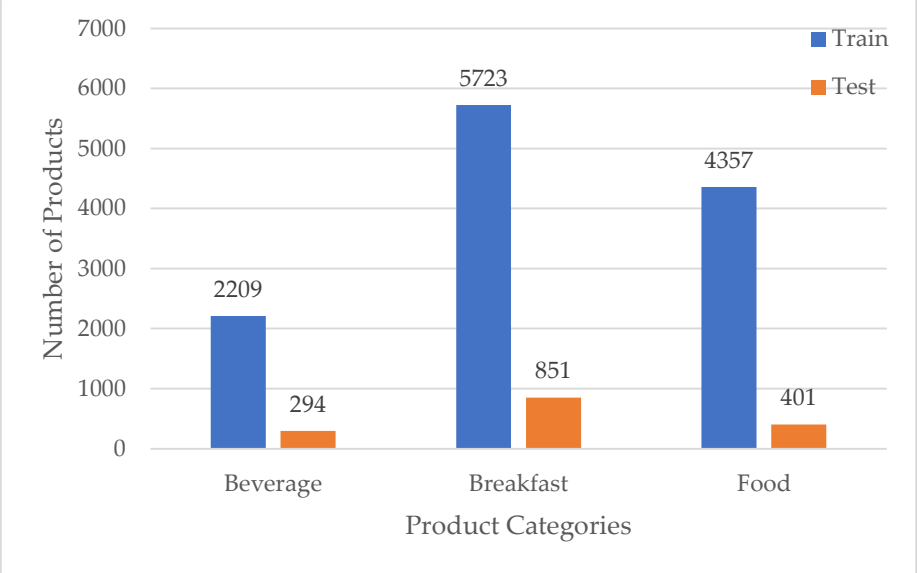


Figure 5.1 Distribution of labeled products

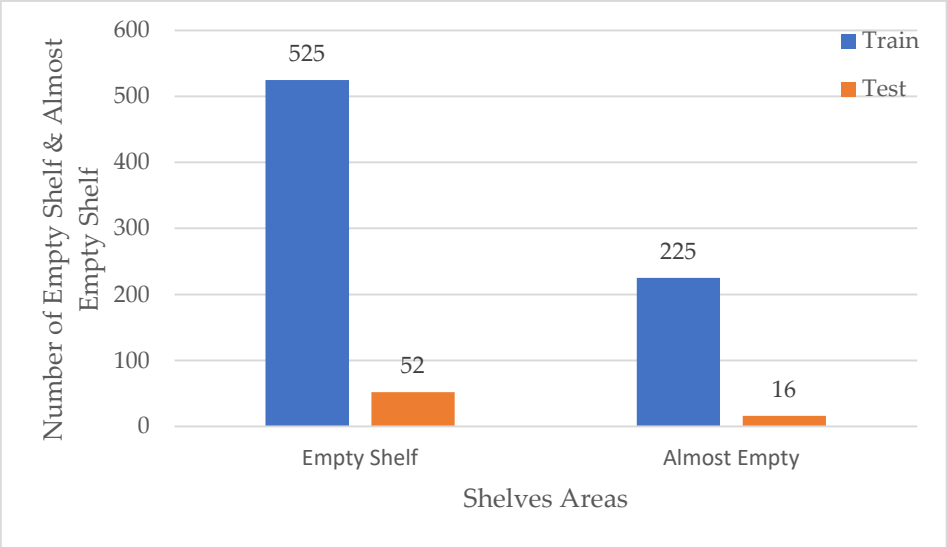


Figure 5.2 Distribution of labeled shelves' areas

In this study, a file was created to store annotation info for each image, and Pascal Visual Object Classes (VOC) was used as an annotation file, which contains all information about the images such as bounding box coordinates and class names. A sample Pascal VOC format is shown in Figure 5.3. Since it is in an XML file format, its format can be easily converted to other formats. Different object detection algorithms were used for different file formats to store labeling information of items on the images. Since products and shelf areas are annotated in a Pascal VOC format, we developed several conversion tools in C-Sharp to convert its format to the required formats.

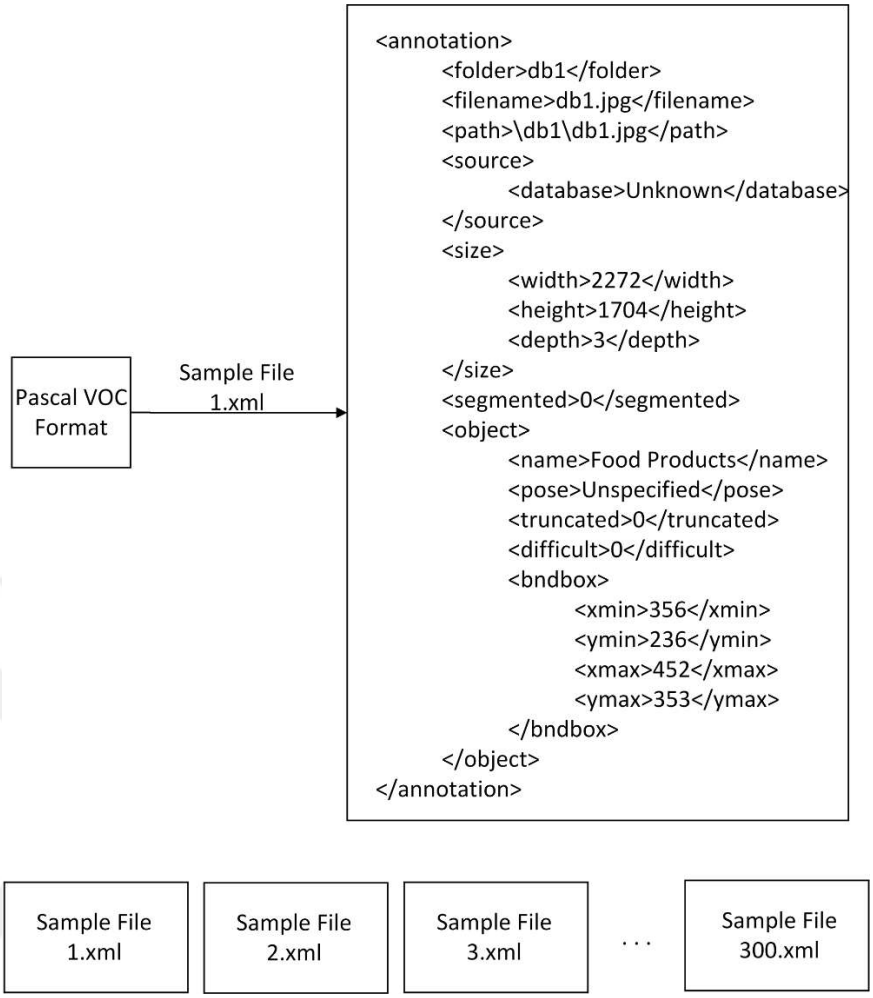


Figure 5.3. Structure of Pascal VOC format for a sample file

For RetinaNet, each file extension was converted from XML to TXT. Sample file structure and conversion details are shown in Figure 5.4. The class labels of the products were inserted in a CSV file. Finally, CSV files for the training and test sets were generated from the Pascal VOC file.

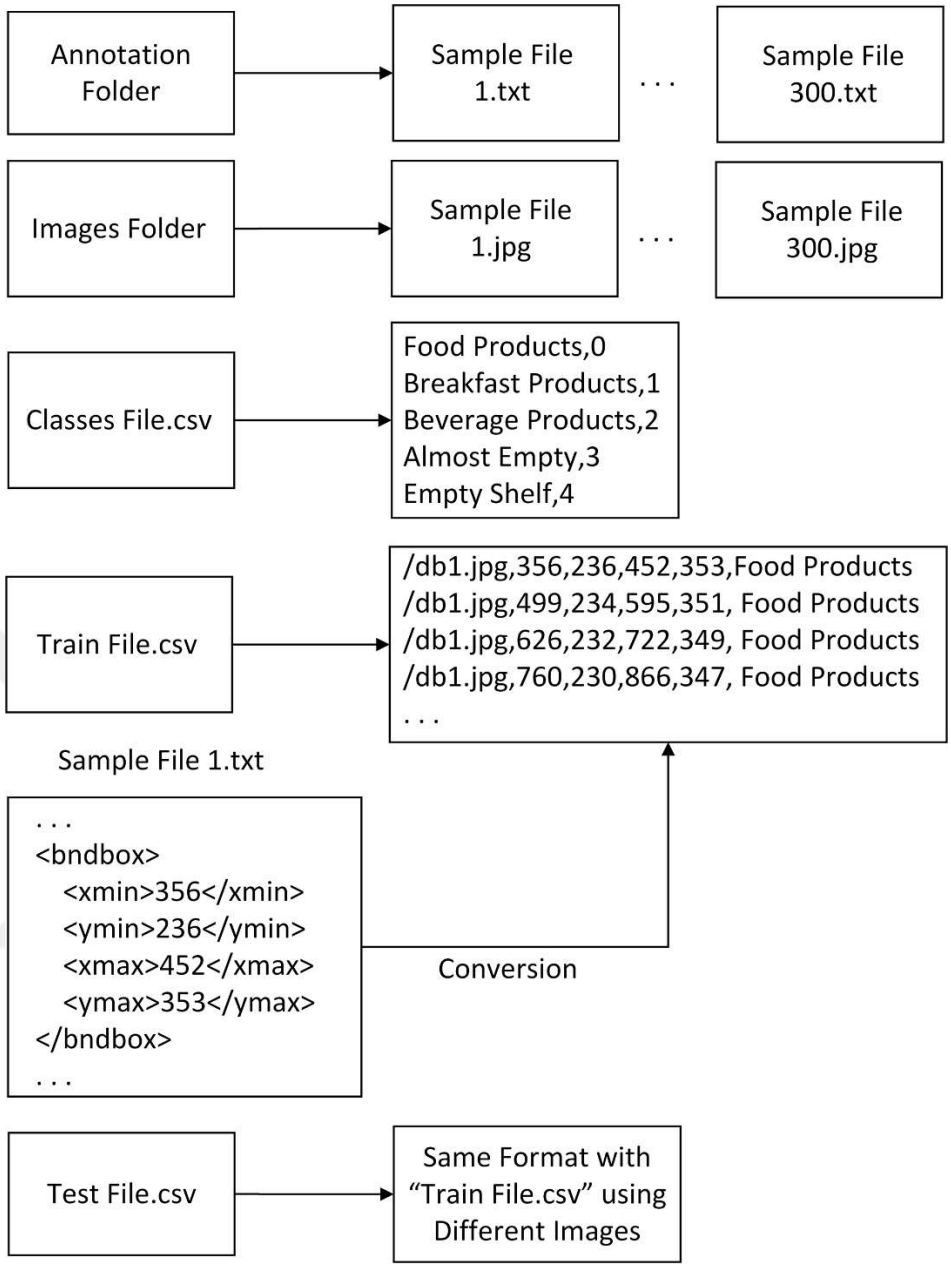


Figure 5.4 RetinaNet sample file structure and conversion details

For YOLOv3 and YOLOv4, each file extension was converted from XML to TXT, and conversion was applied from Pascal VOC to YOLO format. A sample file structure and conversion details are shown in Figure 5.5. The class labels of the products were inserted in a NAMES file. Finally, TXT files for the training and test sets were generated to store the locations of images.

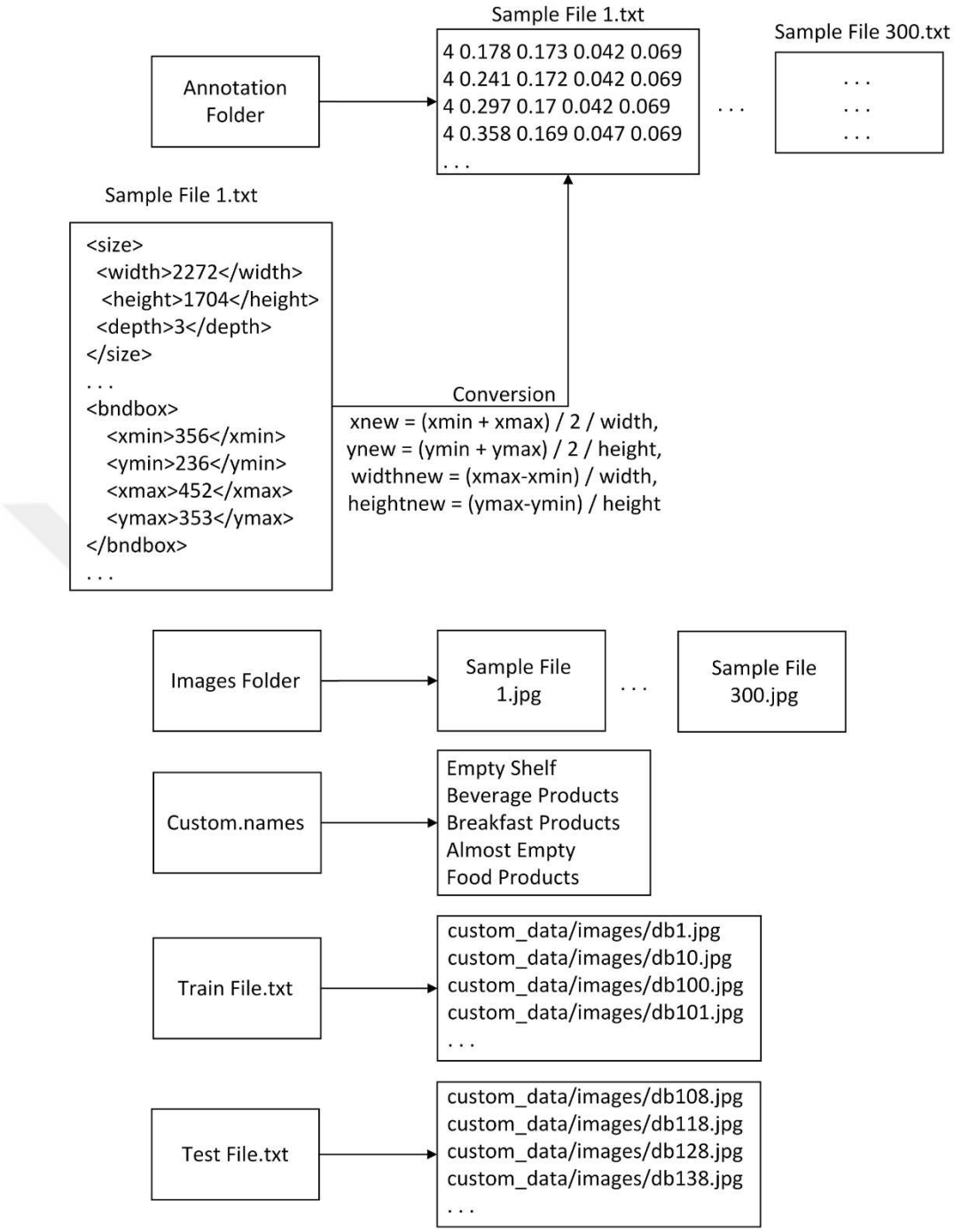


Figure 5.5 YOLOv3 and YOLOv4 sample file structure and conversion details

For evaluating our proposed approach, first of all, a training process was started on the selected three one-stage detectors (RetinaNet, YOLOv3, and YOLOv4) using labeled images. The experimental environment was created on a computer that has the following specifications: Ubuntu 20.04 operating system, GeForce RTX 2080 TI GAMING X TRIO 11GB graphical processing unit (GPU), AMD Ryzen 7 3700X

3.6GHz/4.4GHz processor, and 32GB 3600MHz DDR4 memory. The training operation is processed on the GPU for each detector.

The following open-source frameworks were used in this study: Keras based framework for RetinaNet ("Keras RetinaNet," n.d.), Darknet based framework for YOLOv3 ("YOLOv3 Framework," n.d.), and YOLOv4 ("YOLOv4 Framework," n.d.). For each detector, the following parameters were determined: input image size as 512x512, learning rate as 0.001, and the number of iterations as 10000. RetinaNet was trained using two different convolutional neural networks (CNN) based backbones: ResNet50 and ResNet100. Darknet53, which is a new feature extraction network, was used as the backbone of YOLOv3. For YOLOv4, the CSPDarknet53 backbone was used. This backbone uses Cross Stage Partial Network (CSPNet) approach for partitioning and merging feature maps based on a cross-stage hierarchy (Bochkovskiy et al., 2020). Besides, for each backbone, pre-trained models were used, which are "model50.h5" and "model101.h5" for RetinaNet, "yolov3.weights" for YOLOv3, and "yolov4.conv.137" for YOLOv4.

5.3 Experimental Results

At the end of the first training process, YOLOv4 with CSPDarknet53 had the highest success rates; 0.9187 for mAP, 0.9100 for F1-score, and 0.9600 for Recall. These results were followed by RetinaNet with ResNet101 and ResNet50 backbones, and YOLOv3 with Darknet53 backbone. More details of the training results are shown in Table 5.1. From the results, it can be clearly seen that a significant performance improvement (>20%) was achieved by the proposed method on average. For example; YOLOv4 (0.9616) is significantly better than RetinaNet (0.7245) in the detection of breakfast products. YOLOv4 remarkably outperformed the rest for the "Empty Shelf" and "Almost Empty Shelf" classes with 0.9136 and 0.8125 of average precision values respectively.

Table 5.1 Comparison of success rates of the methods on the labeled images for three different one-stage detectors with four different backbones

Classes	**RetinaNet (Backbone: ResNet50)**	**RetinaNet (Backbone: ResNet101)**	**YOLOv3 (Backbone: Darknet53)**	**YOLOv4 (Backbone: CSPDarkNet53)**
			AP	
Beverage Product	0.9469	0.9625	0.8636	**0.9808**
Breakfast Product	0.6975	0.7245	0.8003	**0.9616**
Food Product	0.8886	0.8697	0.6321	**0.9252**
Empty Shelf	0.8481	0.8387	0.8189	**0.9136**
Almost Empty Shelf	0.2386	0.1561	0.5884	**0.8125**
mAP	0.7239	0.7103	0.7406	**0.9187**
F1-score	0.7333	0.7430	0.6600	**0.9100**
Recall	0.8105	0.8228	0.6600	**0.9600**

In this study, we propose semi-supervised learning on OSA, named SOSA, for the first time to take the advantage of unlabeled data because labeling products on image data is an expensive, tedious, difficult, and time time-consuming process. For this reason, products and shelf areas were labeled on a small number of images to compare our SOSA approach with the standard supervised OSA approach. The one-stage detector that had the highest success rate was selected with their trained model from the first training phase to use for the semi-supervised learning approach. The proposed SOSA method was evaluated on image datasets, with different ratios of labeled samples varying from 20% to 80%. The distribution of samples is shown in Table 5.2. For example, in the first experiment, 300 labeled images and 1200 unlabeled images were used to build a classifier. Firstly, unlabeled images were labeled using the selected one-stage detector with its model constructed in the previous training phase. In this way, a pseudo-labeled image dataset is obtained from an unlabeled image dataset. 49573 products were grouped under three categories and 2145 shelf areas were grouped under two categories for both labeled and pseudo-labeled images.

Table 5.2 Distribution of labeled samples on image datasets

Dataset ID	**Percentage of Labeled Images**	**Number of Labeled Images**	**Number of Unlabeled Images**	**Number of Total Images**
D1	20%	300	1200	1500
D2	40%	300	450	750
D3	60%	300	200	500
D4	80%	300	75	375

Training processes were started for each dataset given in Table 5.2 to verify the effectiveness of the proposed SOSA method. Here, this experiment is performed to test the different ratios of labeled images, varying from 20% to 80% with an increment of 20%. The comparison of loss values obtained during the training phases is shown in Figure 5.6. From the figure, it can be seen that loss values were decreased dramatically during the training phase after two thousand iterations. Very close loss values were obtained for different ratios. The results indicate that a small amount of labeled data is enough to receive similar loss values since the algorithm benefit from unlabeled data. From the results, it can be concluded that a labeled data that contains shelf images taken from a grocery store can be extended with unlabeled images to achieve high accuracy.

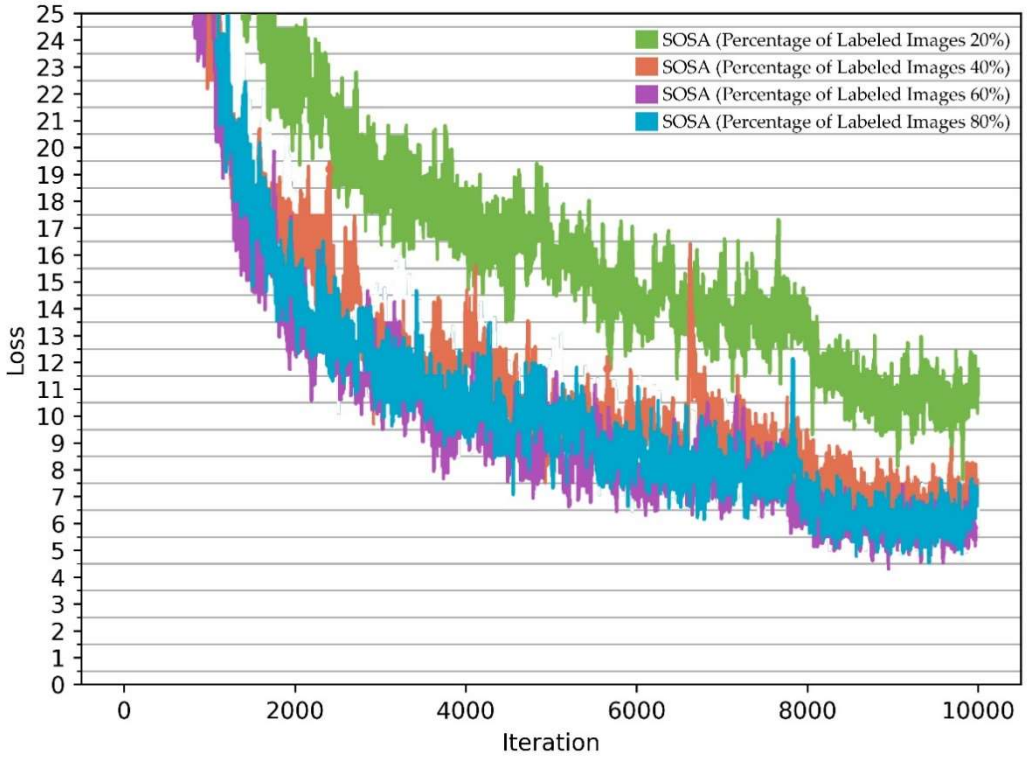


Figure 5.6 Comparison of loss value changes during the training phase

As shown in Figure 5.7, satisfactory results were achieved by the proposed SOSA approach. When the percentage of labeled images was 20%, mAP, F1-score, and Recall values were obtained as 0.7209, 0.8100, and 0.8300, respectively. When the percentage of labeled images was 40%, mAP was calculated as 0.7257, F1-score was achieved by 0.7800, and Recall was measured as 0.7800. When using a 60%-40% labeled-unlabeled ratio, mAP was 0.8622, F1-score was 0.8700, and Recall was 0.9000. When the labeled image rate was 80%, mAP was 0.8927, F1-score was 0.9000, and Recall was 0.9300. As has been observed in Figure 5.7, it is possible to provide good generalization ability for the OSA problem by applying a small number of labeled image data. It also can be seen from the results that an improvement could be expected from our method in circumstances where the ratio of labeled data grows. For example, after adding %20 labeled data, we found that the accuracy of the SOSA method increased from 0.7257 to 0.8622, with approximately 14% improvement. The results indicate that an initial amount of labeled ordinal data can be sufficient enough to learn the model, but additional labeled data can improve the classification performance.

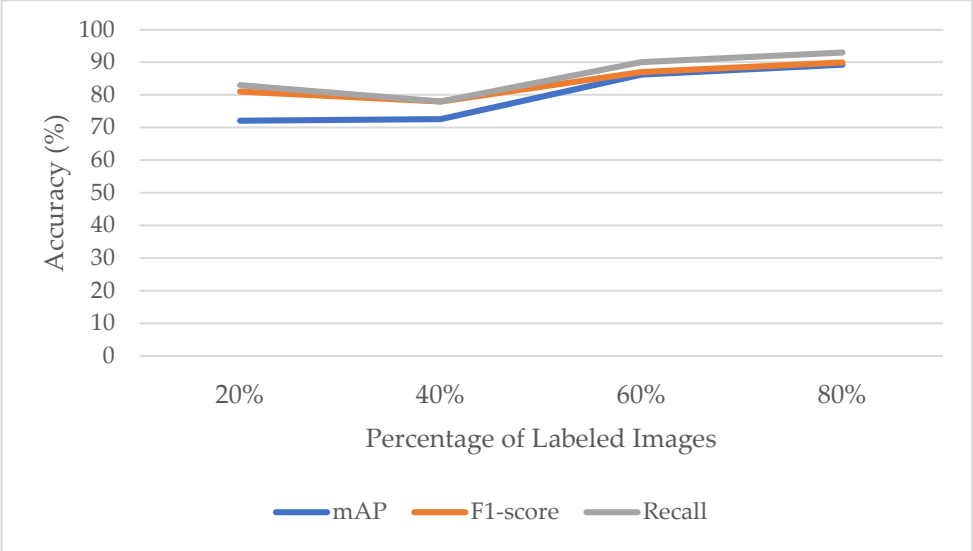
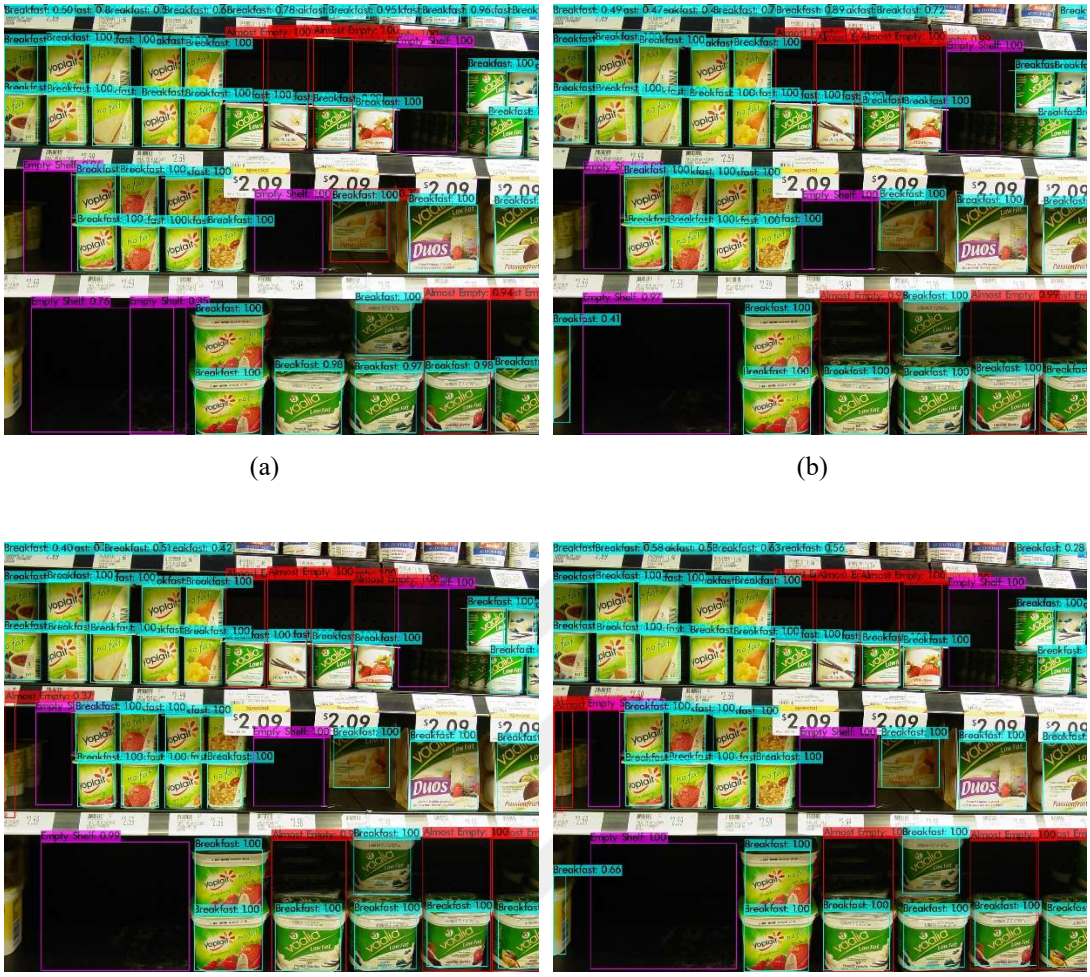


Figure 5.7. Success rates when different ratios of labeled data are considered

The sample object detection results related to the breakfast section are shown in Figure 5.8. Each subfigure shows the results obtained by the models that were trained using labeled samples varying from 20% to 80%. While some almost-empty areas, which are highlighted in red color, could not be detected with a 20%-80% labeled-unlabeled ratio, these areas could be detected with higher labeled data ratios. The detection accuracy of almost-empty areas was increased as the labeled data ratio was increased. Besides, from the SOSA XAI perspective, the results were interpreted in a more understandable way for users. SOSA XAI output message is as follows: “8 almost-empty and 4 empty shelves are existing on the breakfast section. In order to determine the section, the products on the picture are counted and the category that has the most products is selected”.

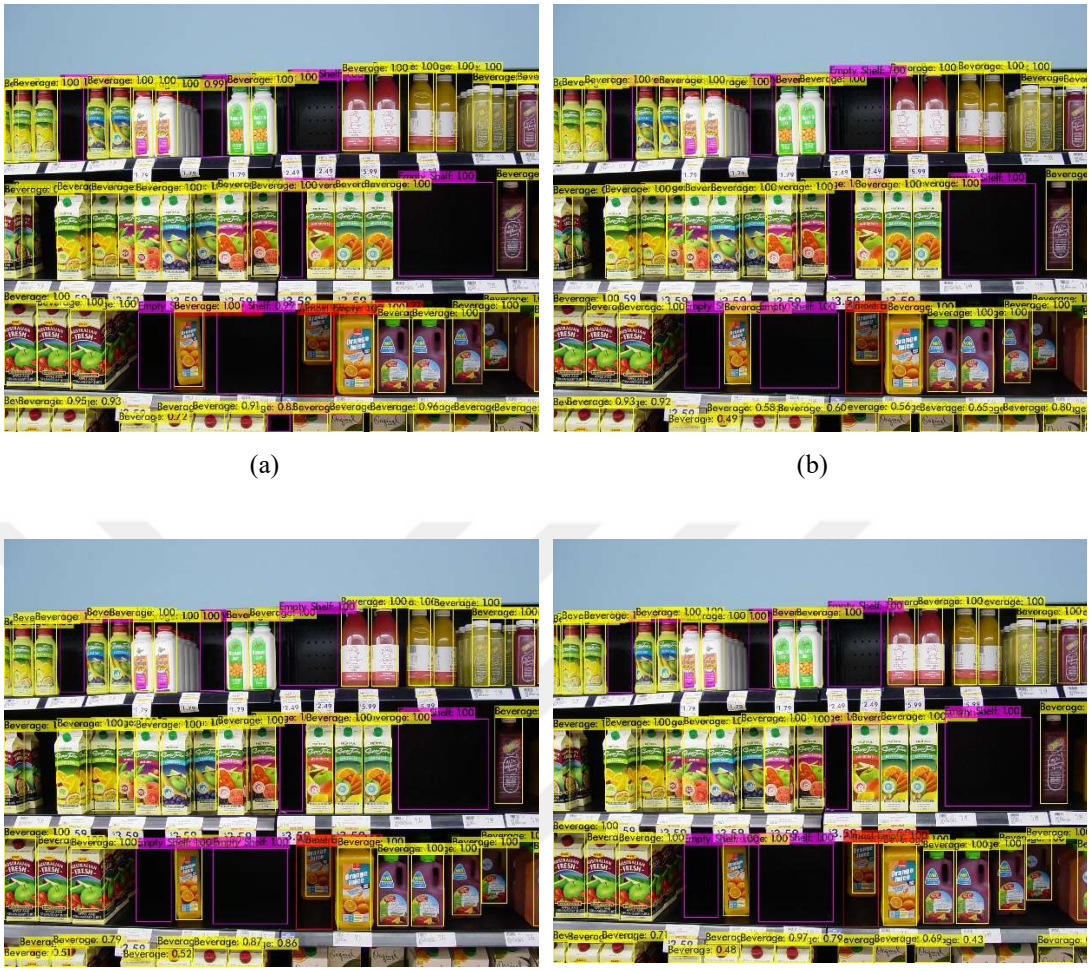


(a) (b)

(c) (d)

Figure 5.8 Sample object detection results of the breakfast section. It shows detection results using the model that was trained with (a) 20% of labeled images; (b) 40% of labeled images; (c) 60% of labeled images; and (d) 80% of labeled images (modified from ("WebMarket is an image database built for Computer Vision Research," n.d.))

For the beverage section, the sample object detection results are shown in Figure 5.9. The sample shelf image has eight empty and one almost-empty shelves. All trained models detected empty shelves (highlighted in violet color) and almost-empty shelf (highlighted in red color) with high probabilities. For example, the model that was trained with 20% labeled images detected one of the empty shelves with 99% probability and the almost-empty shelf with 100% probability. The model that was trained with 80% labeled images detected one of the empty shelves with 100% probability and the almost empty shelf with 100% probability. Besides, all models detected beverage products with high probabilities.



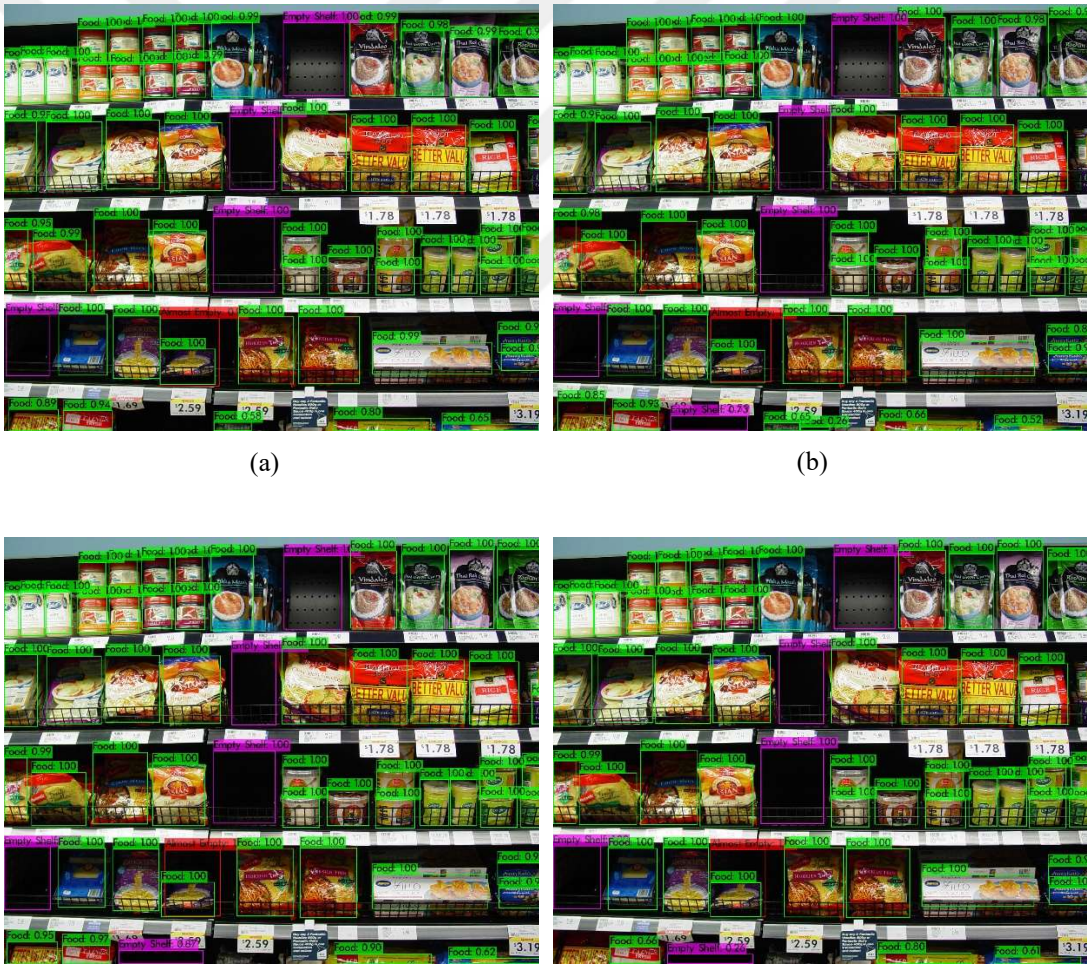
(a) (b)

(c) (d)

Figure 5.9 Sample object detection results of the beverage section. It shows detection results using the model that was trained with (a) 20% of labeled images; (b) 40% of labeled images; (c) 60% of labeled images; and (d) 80% of labeled images (modified from ("WebMarket is an image database built for Computer Vision Research," n.d.))

In this study, we focused on detecting empty and almost-empty areas on the breakfast and the beverage sections. The images that do not include beverage and breakfast products labeled as "Food Product". The SOSA approach was also tested using shelf images that were taken from the food sections. Figure 5.10 shows the sample object detection results for the rice and pasta section. The results showed that products were correctly labeled as "Food Product" with high probabilities. Most of the images in the dataset are frontal views and products in front of the shelves are fundamentally analyzed to be detected in this study. When a product is on the backside

of a shelf, the system generates an almost-empty shelf notification. However, if the depth of a shelf is too deep, the system may detect the products at the backside with low probability. In the cases of empty and almost-empty shelf notifications, a responsible clerk can put new items on the shelf to prevent customer and profit loss. It can also be noted that a package may contain multiple items but it may look like a single object from the outside, and in this case, the package is detected as a single object by the system, as expected. Besides, some products can be damaged. In this case, these products can also be detected by the system, but with lower probability due to the change in their appearance.



(a) (b)

(c) (d)

Figure 5.10 Sample object detection results of the rice and pasta section. It shows detection results using the model that was trained with (a) 20% of labeled images; (b) 40% of labeled images; (c) 60% of

labeled images; and (d) 80% of labeled images (modified from (“WebMarket is an image database built for Computer Vision Research,” n.d.))

The comparison of the proposed SOSA approach with the existing approaches in terms of accuracy is shown in Table 5.3. The experimental results showed that the proposed approach outperformed the existing approaches (RetinaNet and YOLOv3) in terms of accuracy. The best performance was achieved for the breakfast products. When the percentage of labeled images varied between 80% and 20%, the AP values were obtained as 0.9467, 0.9626, 0.9253, and 0.8414, respectively. At most, all the algorithms had difficulty in distinguishing the "Almost Empty" class. Compared to the existing methods, approximately 25% improvement was achieved for the almost-empty class by the proposed method. Besides, this achievement of the SOSA method can be increased by extending the dataset with unlabeled images that contain more almost-empty areas. From the empty-shelf class perspective, satisfactory AP results (>0.8) were obtained by the proposed method. In addition, when AP results were evaluated based on product classes, it was seen that AP values increased compared to the previous methods on average. Based on the results, it can be concluded that the proposed SOSA method has good generalization ability in distinguishing all the classes, so it can be effectively used to recognize them well.

Table 5.3 Comparison of accuracies between SOSA and existing approaches

Classes	**RetinaNet (Backbone: ResNet50)**	**RetinaNet (Backbone: ResNet101)**	**YOLOv3 (Backbone: Darknet53)**	**SOSA (80% Labeled Images)**	**SOSA (60% Labeled Images)**	**SOSA (40% Labeled Images)**	**SOSA (20% Labeled Images)**
			AP				
Beverage Product	0.9469	0.9625	0.8636	0.9586	0.8797	0.7477	0.8224
Breakfast Product	0.6975	0.7245	0.8003	0.9467	0.9626	0.9253	0.8414
Food Product	0.8886	0.8697	0.6321	0.9303	0.8680	0.7308	0.8718
Empty Shelf	0.8481	0.8387	0.8189	0.8410	0.8736	0.7601	0.7216
Almost Empty Shelf	0.2386	0.1561	0.5884	0.7866	0.7269	0.4646	0.3471
mAP	0.7239	0.7103	0.7406	0.8927	0.8622	0.7257	0.7209
F1-score	0.7333	0.7430	0.6600	0.9000	0.8700	0.7800	0.8100
Recall	0.8105	0.8228	0.6600	0.9300	0.9000	0.7800	0.8300

CHAPTER SIX
DISCUSSION

Providing high on-shelf-availability is a key factor to increase profits in grocery stores. For this purpose, this study is the first attempt to combine two concepts "semi-supervised learning" and "on-shelf availability" (SOSA). The proposed SOSA method aims to decrease out-of-stock by automatically detecting and classifying products using shelf images based on the semi-supervised learning principle. The main purposes of the SOSA method are to decrease the need for manual image labeling, to obtain satisfactory results using a small amount of manually labeled images, and to provide additional knowledge present in unlabeled image data.

SOSA builds a classification model that allows the detection of “Product”, “Empty Shelf”, and “Almost-Empty Shelf” classes by using both labeled and unlabeled image data. When building the model, three different deep learning techniques are used: RetinaNet, YOLOv3, andYOLOv4. This study shows that the proposed approach improves accuracy when monitoring OSA.

In the experimental studies, the effectiveness of the proposed SOSA method was demonstrated on a real-world image dataset, with different ratios of labeled samples varying from 20% to 80%. As can be seen from Table 5.3 that the proposed SOSA approach outperformed the previous approaches on average. For instance, according to the experimental results, 15%-18% improvement is achieved on average accuracy. When developing an OSA system, some challenges could be encountered and overcome. The first one is that some products can lie on the shelf or their packages can be damaged. In these cases, these kinds of products can be detected with low probability due to the change in their appearance. To increase detection probability, the images containing this type of product can be included in the training set. In this way, the model can learn from these kinds of objects, and therefore these products can be detected with higher accuracies. Another difficulty is that some objects can stand in front of the shelf and they can prevent detecting products on the shelf. These objects can be shoppers, employees, grocery store trolleys, or something else. In this study,

we implemented a solution to check the presence of people on the image. If a person is detected, this image is discarded by the system, and then a new image is taken after a period of time. This issue is also important to protect the privacy of customers. Similar solutions can also be implemented for other objects.

The SOSA method is powered by the SOSA XAI application. SOSA XAI was developed to provide transparency and understandability of the OSA model for the users. Thanks to this tool, detection results can be interpreted more easily. Furthermore, it provides insight into what the OSA model is paying attention to. In addition, it provides evidence for the results, and validity of the process. Moreover, the training set can be enhanced in a proper manner to adopt the application for changes without an expert in the AI field. Besides, SOSA AR mobile application was developed to monitor notifications and last audits and it was integrated to SOSA and SOSA XAI.

The main findings of this study can be listed as follows. (1) It was observed that "semi-supervised learning" provides many advantages for monitoring OSA, including improving efficiency, reducing labeling cost, providing additional knowledge present in unlabeled data, and increasing the applicability of machine learning in the retail sector. (2) The combination of three deep learning techniques (RetinaNet, YOLOv3, YOLOv4) improves accuracy when monitoring OSA. (3) Explainable AI is a powerful tool in monitoring OSA since it provides users with an explanation of individual decisions and enables users to manage, understand, and trust the OSA model. (4) The proposed SOSA method has the potential to expand the application of machine learning in grocery stores, thanks to its advantages. (5) The SOSA AR mobile application provides easier monitoring to users using augmented reality technology.

CHAPTER SEVEN
CONCLUSIONS AND FUTURE WORK

7.1 Conclusions

Providing high on-shelf availability is a key factor to increase profits in grocery stores. For this purpose, the traditional OSA applications use the labeled image data when building a classifier. However, a large amount of data generated in real-life is unlabeled, and manually labeling products on the images is an expensive process. To overcome this problem, this thesis proposes a new approach, called SOSA, which combines two concepts "semi-supervised learning" and "on-shelf availability" for the first time. The proposed SOSA method addresses the problem by automatically allowing the model to integrate the available unlabeled image data with very little cost. Our proposed method detects empty and almost empty shelves based on each product category using a small amount of labeled data and a large amount of unlabeled data.
It is the first time that YOLOv4 object detection architecture is used for monitoring OSA. This study is also original in that it compares three deep learning approaches (RetinaNet, YOLOv3, YOLOv4) for monitoring OSA in the retail sector.

The experiments were conducted on image datasets with different ratios of labeled samples varying from 20% to 80%. The experimental results show that the proposed approach outperforms the existing approaches (RetinaNet and YOLOv3) in terms of accuracy.

Moreover, this thesis presents the first demonstration of explainable artificial intelligence (XAI) on OSA, called SOSA XAI. Thanks to the SOSA XAI, users can understand, trust, and manage the proposed deep learning model and modify the dataset to adapt the deep learning model for changes. Also, an augmented reality (AR) mobile application is developed to monitor the latest status of the shelf, called SOSA AR, and thanks to the SOSA AR, users can monitor notifications and last audits easily.

7.2 Future Work

Currently, the developed SOSA XAI software application uses rule-based interpretation to explain the outputs of the proposed approach. Instead of rule-based interpretation, a deep learning-based interpretation can be developed in the future. Furthermore, new features can be developed such as planogram checking and managing using augmented reality technology for the SOSA AR application. These new features can allow to track that products are in right place on the shelf or not and store clerks can be notified. Moreover, more than three one-stage detectors can be integrated into the SOSA approach as future work. Our proposed approach is not limited to grocery stores. Also, it can be applied in different areas such as pharmacy and warehouses by changing initial labeled images and determining classes according to the related field.

REFERENCES

Angeles, R. (2005). Rfid technologies: supply-chain applications and implementation issues. *Information Systems Management*, *22* (1), 51–65.

Azuma, R. T. (1997). A survey of augmented reality. *Teleoperators and Virtual Environments*, *6* (4), 355–385.

Bargoti, S., & Underwood, J. (2017). Deep fruit detection in orchards. *Proceedings - IEEE International Conference on Robotics and Automation*, 3626–3633.

Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., et al. (2020). Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, *58*, 82–115.

Bay, H., Ess, A., Tuytelaars, T., & Van Gool, L. (2008). Speeded-up robust features (SURF). *Computer Vision and Image Understanding*, *110* (3), 346–359.

Bayle, A., Laurent, G., & Macé, S. (2006). Assessing the frequency and causes of out-of-stock events through store scanner data. *Paris: Paris Chambre de Commerce et d'Industrie de Paris*.

Bo, L., Ren, X., & Fox, D. (2013). Unsupervised feature learning for RGB-D based object recognition. *The 13th International Symposium on Experimental Robotics*, 387–402.

Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). *YOLOv4: optimal speed and accuracy of object detection*. Retrieved July 20, 2020, from http://arxiv.org/abs/2004.10934

Bottani, E., Bertolini, M., Rizzi, A., & Romagnoli, G. (2017). Monitoring on-shelf availability, out-of-stock and product freshness through RFID in the fresh food supply chain. *International Journal of RF Technologies: Research and Applications*, *8* (1–2), 33–55.

Cheng, J., Yang, J., & Zhou, Y. (2005). A novel adaptive Gaussian mixture model for background subtraction. *Lecture Notes in Computer Science*, *3522* (I), 587–593.

Confalonieri, R., Coba, L., Wagner, B., & Besold, T. R. (2020). A historical perspective of explainable artificial intelligence. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1–21.

Corsten, D., & Gruen, T. (2003). Desperately seeking shelf availability: An examination of the extent, the causes, and the efforts to address retail out-of-stocks. *International Journal of Retail & Distribution Management*, *31* (12), 605–617.

Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., & Darrell, T. (2014). DeCAF: A deep convolutional activation feature for generic visual recognition. *31st International Conference on Machine Learning, ICML 2014*, *2*, 988–996.

Ehrenthal, J. C. F., Gruen, T. W., & Hofstetter, J. S. (2014). Value attenuation and retail out-of-stocks: a service-dominant logic perspective. *International Journal of Physical Distribution and Logistics Management*, *44* (1), 39–57.

Eitel, A., Springenberg, J. T., Spinello, L., Riedmiller, M., & Burgard, W. (2015). Multimodal deep learning for robust RGB-D object recognition. *IEEE International Conference on Intelligent Robots and Systems*, 681–687.

Emmert-Streib, F., Yli-Harja, O., & Dehmer, M. (2020). Explainable artificial intelligence and machine learning: a reality rooted perspective. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *10* (6), 1–8.

García-Arca, J., Prado-Prado, J. C., & González-Portela Garrido, A. T. (2020). On-shelf availability and logistics rationalization. A participative methodology for supply chain improvement. *Journal of Retailing and Consumer Services*, *52* (July 2019), 101889.

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 580–587.

Gruen, T. W., & Corsten, D. (2008). *A comprehensive guide to retail out-of-stock reduction in the fast-moving consumer goods industry.* Retrieved May 12, 2020, from http://www.nacds.org/pdfs/membership/out_of_stock.pdf.

Gruen, T. W., Corsten, D. S., & Bharadwaj, S. (2002). Retail out-of-stocks: a worldwide examination of extent causes and consumer responses. *Grocery Manufacturers of America*.

Higa, K., & Iwamoto, K. (2018). Robust estimation of product amount on store shelves from a surveillance camera for improving on-shelf availability. *IST 2018 - IEEE International Conference on Imaging Systems and Techniques, Proceedings*, 1–6.

Higa, K., & Iwamoto, K. (2019). Robust shelf monitoring using supervised learning for improving on-shelf availability in retail stores. *Sensors*, *19* (12), 2722.

Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2261–2269.

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *32nd International Conference on Machine Learning, ICML 2015*, *1*, 448–456.

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., et al. (2014). Caffe: Convolutional architecture for fast feature embedding. *MM 2014 - Proceedings of the 2014 ACM Conference on Multimedia*, 675–678.

Judith, B., Sigal, L., Mori, G., & Mandt, S. (2018). Informed priors for deep representation learning. *1st Symposium on Advances in Approximate Bayesian Inference*, 1–5.

Jund, P., Abdo, N., Eitel, A., & Burgard, W. (2016). *The freiburg groceries dataset*. Retrieved January 20, 2020, from http://arxiv.org/abs/1611.05799.

Kajabad, E. N., & Ivanov, S. V. (2019). People detection and finding attractive areas by the use of movement detection analysis and deep learning approach. *Procedia Computer Science*, *156*, 327–337.

Kejriwal, N., Garg, S., & Kumar, S. (2015). Product counting using images with application to robot-based retail stock assessment. *IEEE Conference on Technologies for Practical Robot Applications, TePRA*, 1–6.

Keras retinanet. (n.d.). Retrieved November 11, 2020, from https://github.com/fizyr/keras-retinanet

Klasson, M., Zhang, C., & Kjellström, H. (2019). A hierarchical grocery store image dataset with visual and semantic labels. *Proceedings - 2019 IEEE Winter Conference on Applications of Computer Vision, WACV 2019*, *2*, 491–500.

Krizhevsky, A. (2009). *Learning multiple layers of features from tiny images*. Retrieved January 01, 2020, from https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf.

Krizhevsky, A., Sutskever, I., & E. Hinton, G. (2012). Imagenet classification with deep convolutional neural networks. *In Advances in Neural Information Processing Systems*.

Labelimg. (n.d.). Retrieved November 11, 2020, from https://github.com/tzutalin/labelImg.

Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2017). Feature pyramid networks for object detection. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 936–944.

Lin, T. Y., Goyal, P., Girshick, R., He, K., & Dollar, P. (2018). Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *42* (2), 318–327.

Liu, S., Li, W., Davis, S., Ritz, C., & Tian, H. (2016). Planogram compliance checking based on detection of recurring patterns. *IEEE Multimedia*, *23* (2), 54–63.

Liu, S., & Tian, H. (2016). Planogram compliance checking using recurring patterns. *Proceedings - 2015 IEEE International Symposium on Multimedia, ISM 2015*, 27–32.

Low, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 91–110.

Merler, M., Galleguillos, C., & Belongie, S. (2007). Recognizing groceries in situ using in vitro training data. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition,* 1-8.

Michael, K., & McCathie, L. (2005). The pros and cons of RFID in supply chain management. *4th Annual International Conference on Mobile Business, ICMB 2005*, 623–629.

Moorthy, R., Behera, S., & Verma, S. (2015). On-shelf availability in retailing. *International Journal of Computer Applications*, *115*(23), 47–51.

Moorthy, R., Behera, S., Verma, S., Bhargave, S., & Ramanathan, P. (2015). Applying image processing for detecting on-shelf availability and product positioning in retail stores. *ACM International Conference Proceeding Series*, 451–457.

Morimoto, T., Kiriyama, O., Harada, Y., Adachi, H., Koide, T., & Mattausch, H. J. (2005). Object tracking in video pictures based on image segmentation and pattern matching. *Proceedings - IEEE International Symposium on Circuits and Systems*, 3215–3218.

Mureşan, H., & Oltean, M. (2018). Fruit recognition from images using deep learning. *Acta Universitatis Sapientiae, Informatica*, *10* (1), 26–42.

Musalem, A., Olivares, M., Bradlow, E. T., Terwiesch, C., & Corsten, D. (2010). Structural estimation of the effect of out-of-stocks. *Management Science*, *56* (7), 1180–1197.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., & Ng, A. Y. (2011). Reading digits in natural images with unsupervised feature learning. *NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 1-9.

Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., & Ng, A. Y. (2011). Multimodal deep learning. *Proceedings of the 28th International Conference on Machine Learning, ICML 2011*, 689–696.

Oquab, M., Bottou, L., Laptev, I., & Sivic, J. (2014). Learning and transferring mid-level image representations using convolutional neural networks. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1717–1724.

Papakiriakopoulos, D. A. (n.d.). Automatic detection of out-of-shelf products in the retail sector supply chain. *Complexity*, 1–8.

Prakash, C. D., & Karam, L. J. (2019). *It GAN DO better: GAN-based detection of objects on images with varying quality*. Retrieved February 17, 2021, from https://arxiv.org/abs/1912.01707.

Radford, A., Metz, L., & Chintala, S. (2016). Unsupervised representation learning with deep convolutional generative adversarial networks. *4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings*, 1–16.

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 779–788.

Redmon, J., & Farhadi, A. (2018). *YOLOv3: An incremental improvement*. Retrieved November 12, 2020, from http://arxiv.org/abs/1804.02767.

Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *39* (6), 1137–1149.

Rosado, L., Goncalves, J., Costa, J., Ribeiro, D., & Soares, F. (2016). Supervised learning for out-of-stock detection in panoramas of retail shelves. *IST 2016 - 2016 IEEE International Conference on Imaging Systems and Techniques, Proceedings*, *2020* (6389), 406–411.

Sa, I., Ge, Z., Dayoub, F., Upcroft, B., Perez, T., & McCool, C. (2016). Deepfruits: A fruit detection system using deep neural networks. *Sensors*, *16* (8), 1222.

Santra, B., & Mukherjee, D. P. (2019). A comprehensive survey on computer vision based approaches for automatic identification of products in retail store. *Image and Vision Computing*, *86*, 45–63.

Saran, A., Hassan, E., & Maurya, A. K. (2015). Robust visual analysis for planogram compliance problem. *Proceedings of the 14th IAPR International Conference on Machine Vision Applications, MVA 2015*, 576–579.

Satapathy, R., Prahlad, S., & Kaulgud, V. (2015). Smart shelfie - internet of shelves: For higher on-shelf availability. *Proceedings - 2015 IEEE Region 10 Symposium, TENSYMP 2015*, 70–73.

Senior, A. W., Brown, L., Hampapur, A., Shu, C. F., Zhai, Y., Feris, R. S., et al. (2007). Video analytics for retail. *2007 IEEE Conference on Advanced Video and Signal Based Surveillance, AVSS 2007 Proceedings*, 423–428.

Vargheese, R., & Dahir, H. (2014). An IoT/IoE enabled architecture framework for precision on shelf availability: Enhancing proactive shopper experience. *Proceedings - 2014 IEEE International Conference on Big Data, IEEE Big Data 2014*, 21–26.

Wang, W., Hong, W., Wang, F., & Yu, J. (2020). GAN-knowledge distillation for one-stage object detection. *IEEE Access*, *8*, 60719–60727.

WebMarket is an image database built for computer vision research. (n.d.). Retrieved November 11, 2020, from http://yuhang.rsise.anu.edu.au

Yilmazer, R., & Birant, D. (2021a). Monitoring shelf availability using mobile augmented reality. *International Symposium of Scientific Research and Innovative Studies, ISSRIS'21*. Retrieved February 24, 2021, from https://www.issris.org

Yilmazer, R., & Birant, D. (2021b). Shelf auditing based on image classification using semi-supervised deep learning to increase on-shelf availability in grocery stores. *Sensors*, *21* (2), 327.

YOLOv3 framework. (n.d.). Retrieved November 11, 2020, from https://pjreddie.com/darknet/yolo

YOLOv4 framework. (n.d.). Retrieved November 11, 2020, from https://github.com/AlexeyAB/darknet

Zhang, N., Donahue, J., Girshick, R., & Darrell, T. (2014). Part-based R-CNNs for fine-grained category detection. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *8689 LNCS*(PART 1), 834–849.

Zhang, Y., Wang, L., Hartley, R., & Li, H. (2007). Where's the weet-bix ? *ACCV'07: Proceedings of the 8th Asian Conference on Computer Vision*, *Part 1*, 800–810.

Zhang, Y., Wang, L., Hartley, R., & Li, H. (2009). Handling significant scale difference for object retrieval in a supermarket. *DICTA 2009 - Digital Image Computing: Techniques and Applications*, 468–475.

Zhao, J., Gong, M., Liu, J., & Jiao, L. (2014). Deep learning to classify difference image for image change detection. *Proceedings of the International Joint Conference on Neural Networks*, 411–417.

Zhu, X. (2017). Semi-supervised learning, encyclopedia of machine learning and data mining. In *Encyclopedia of Machine Learning and Data Mining,* 3, 1142-1147.