# 博士学位論文

論文題目　A Study on Flexible Recognition Systems for Customer Activity in Retail Environments

（小売店舗における消費者行動の柔軟な認識システムに関する研究）

提　出　者　　東北大学大学院情報科学研究科

応用情報科学　専　攻

学籍番号　C0ID4002

氏　名　　Jiahao Wen

令和４年度　博士学位論文

# A Study on Flexible Recognition Systems for Customer Activity in Retail Environments

# (小売店舗における消費者行動の柔軟な認識システムに関する研究)

東北大学大学院情報科学研究科 応用情報科学専攻

博士課程後期３年の課程

情報ネットワーク論講座(菅沼・水木研究室)

C0ID4002　Jiahao Wen

2023年１月

# Contents

# List of Figures

# List of Tables

# 1 Introduction

## 1.1 Background

### 1.1.1 Industry 4.0 & Smart Retail

Human industry has undergone three revolutions. Steam power, electricity, computers, and automation mark the three industrial revolutions. Nowadays, we are experiencing the fourth industrial revolution, also called Industry 4.0 [47], which is represented by new technologies, including the Internet of Things (IoT), cloud computing and analytics, artificial intelligence (AI), and machine learning. These technologies are revolutionizing the way companies manufacture, improve, and distribute their products [20]. Smart retail is one of the typical applications of Industry 4.0. In recent years, traditional retail stores have suffered a great shock from the widespread adoption of online shopping. To gain market growth, smart retail is being introduced by more and more retailers. The smart retail is defined as an integration of IoT and data analytics for retail purposes [13, 18], such as store management (e.g. layout optimization, inventory management), market planning (e.g. product promotion) [31], and security (e.g. shoplifting prevention). Figure 1.1 illustrates the common flow of how smart retail works. Various in-store sensors are installed to collect data from the retail store [29]. A recognition system that contains methods to recognize various information from sensor data [18], such as customer route, customer activity, and customer personas. Then, the information is analyzed to support various retail purposes [18], like marketing. Finally, some applications are applied to the retail store for these retail purposes.

As an example, in-store cameras are installed to capture video data about customers' shop-

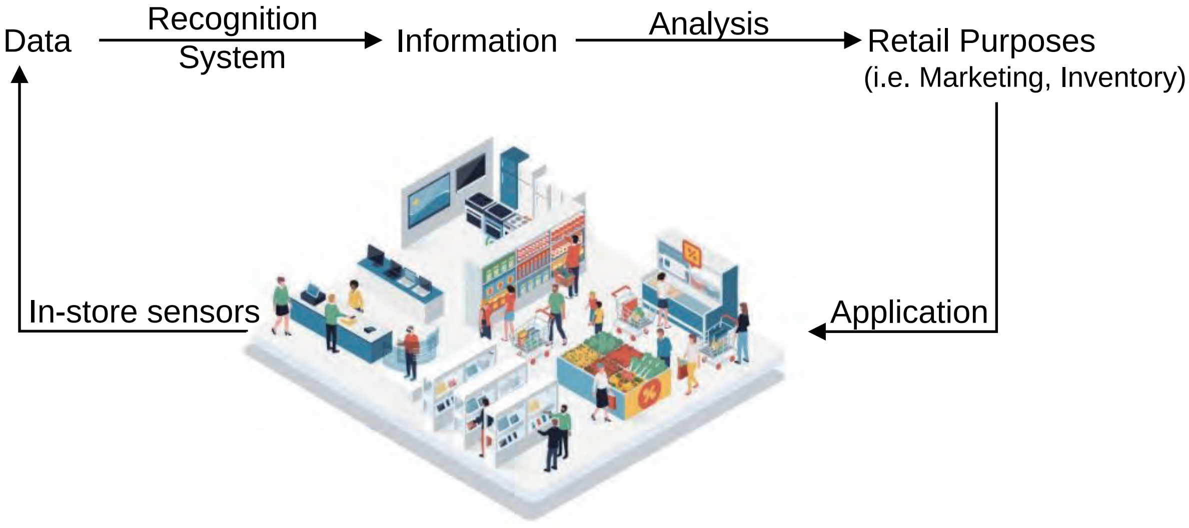


Figure 1.1: Smart retail

ping processes. Then, the recognition system recognizes from the video that most customers return product A back to the shelf and pick up product B next to A. After analysis of this information, it indicates that most customers are more interested in product B. As a result, the retailer reduces the stock of product A and increases the stock of product B. This example demonstrates that determining what information the recognition system should provide is important for smart retail.

### 1.1.2 Customer Activity (CA)

We summarized the most important information in smart retail, named Customer Activity (CA). CA refers to various types of customer's situations in store spaces, including the customer's location, moving route, and behavior. There are many types of CAs and they can be classified into several levels by their abstraction degrees of information. Figure 1.2 shows an example of CA levels. The lowest level is Position which comprises the customer's location. Then, the location is abstracted into the customer's moving route that belongs to a higher CA level Trajectory. To reach a higher CA level, moving routes are abstracted into customer behav-

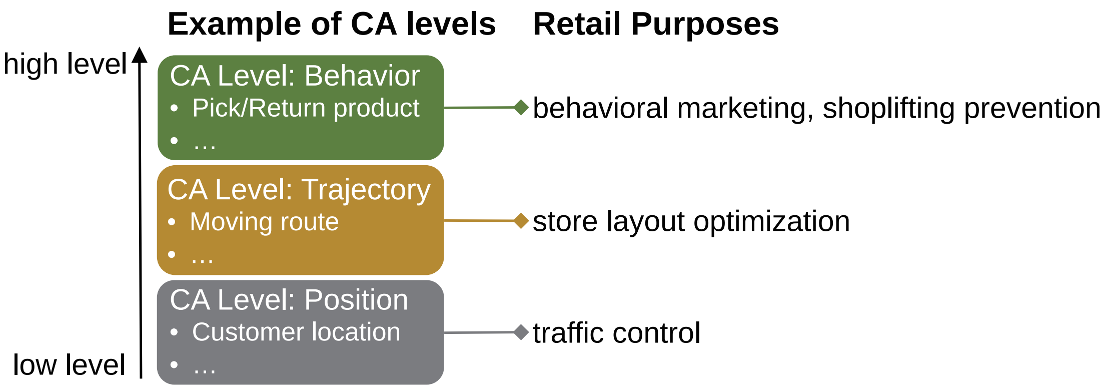


Figure 1.2: Example of CA levels

ior.

Which levels and types of CAs should be recognized using what CA Recognition (CAR) models varies greatly depending on retail purposes and environments. In this study, the collection of CAR models is considered as a CAR system. A CAR system should provide accurate CAs to make sure the information is reliable. Furthermore, it should be able to adapt flexibly to changes in target CAs depending on retail purposes and changes in retail environments. This is because changes in target CAs and retail environments can occur quite frequently, as described below:

**(C1) Target Change** CA has lots of types. Generally, some CAs are chosen as the recognition targets. Target CAs, especially customer behavior (CB), often change with different retail purposes. For instance, the target CA may change depending on the purpose, such as shoplifting prevention and decision-making analysis.

**(C2) Environment Change** The environment refers to in-store implementations for retail service or management, including product types, surveillance cameras, and store layout. The environment usually changes for needs like different store types, festival promotion, inventory supply, security and so on.

Therefore, a CAR system must not only realize accurate CAR but also be flexible enough to adapt to (C1) and (C2). However, existing CAR systems leveraged machine learning (ML)-based models because of their remarkable recognition accuracy. Since they are specialized to their assumed retail environments and target CAs, the adaptation to (C1) and (C2) is difficult due to the necessary re-collecting data and re-training models. This results in substantial adaptation costs for smart retail. Consequently, it is necessary to study a flexible CAR system to adapt to (C1) and (C2).

## 1.2 Research Overview

### 1.2.1 Objective

The main objective of this study is to achieve a flexible CAR system. To achieve flexibility, the CAR system is required to adapt to (C1) Target Change and (C2) Environment Change.

### 1.2.2 Main Issues

To achieve the objective, the main issues are summarized as follows:

**(P1) Hard to Adapt to (C1) Target Change**

As introduced in (C1), target CA, especially target CB, may change because of different retail purposes. Customer-centric retail usually demands different CBs to analyze the customer decision-making process. For CB "pick a product," the retailer could require the discrimination of it to know whether a customer is picking a product with one hand or both hands. This provides information about the customer's effort to pick a product, which could directly influence one's shopping experience. We conclude the reason that causes (P1) as follows:

**(P1-A) High cost of rebuilding CAR models**

As existing ML-based CAR models are specialized for their assumed target CBs, their models are usually no longer usable for the changed target CBs. In that case, the adaptation to (C1) requires high cost of rebuilding CAR models. This results in (P1) Hard to Adapt to (C1) Target Change.

### (P2) Hard to Adapt to (C2) Environment Change

The environment refers to the in-store implementations, e.g. changing the store layout for festival promotion and introducing new products. Existing approaches share problems in the following aspects to adapt to (C2):

#### (P2-A) Reducing performance when increasing CAR models

The complete CAR model is a model that integrates all the necessary processes for the recognition of a particular level of CA from the video input. Existing CAR systems utilize complete CAR models for the required levels of CAs. Since CAs are classified into levels by their abstraction degrees of information, high level CA is supposed to be the abstraction of several low level CAs. Therefore, complete CAR models for high level CA and low level CA must have overlapped processes. When environment changes need to add CAR model(s), e.g. introducing new products leads to adding CAR model(s) to detect the new products, the overlapped processes lead to reducing performance of the CAR system with increasing CAR models. This causes (P2) Hard to Adapt to (C2) Environment Change.

#### (P2-B) Modifications for all CAR models when changing inputs

The complete CAR models in existing CAR systems share the same inputs. This results in high coupling between inputs and CAR models. If inputs change due to environmental changes, such as changing the camera to a new one with a different resolution and view, modifications for all CAR models are inevitable. This leads to (P2) Hard to Adapt to (C2) Environment Change.

### 1.2.3 Contributions

To solve the issues mentioned above, this study contributes to the research objective as follows:

**(S1) Primitive-based Customer Behavior Recognition (CBR) Method**

To deal with (P1), we focus on dealing with the change in target CB, which is a kind of CA. We propose a primitive-based CBR method. CBs are recognized by combination of primitives, each of which is defined as an object's motion or two objects' relation for each frame. Thus, primitives' combination represents the occurrence order of motion/relation changes of object(s) during a CB. In the proposed method, we identify primitives from bounding box trajectories of objects extracted from the video input, and then, match the identified primitives with CBs that are predefined by primitives' combinations to recognize CBs. The primitive-based CBR method handles the problem as below:

**(S1-A) Low cost of changing combination of primitives**

To solve (P1-A), the cost of adaptation to target CB changes is reduced by processes of changing combination of primitives. Compared to the necessary adaptation process of rebuilding CBR models for existing CBR methods, the processes defining new target CBs by primitivesćombination of our proposed method spent less costs. This allows the proposed CBR method to solve (P1-A).

**(S2) Hierarchy-based Customer Activity Recognition (CAR) System**

To solve (P2), we proposed a CAR system based on a hierarchy that organizes CA types into different CA levels from lowest to highest. Each level contains a partial CAR model to recognize particular types of CA with the outputs from its lower level. Therefore, the hierarchy handles the specific problems as below:

**(S2-A) Slight influence when increase CAR models**

To solve (P2-A), complete CAR models are divided into partial CAR models corresponding to the CA levels. Each level includes a partial CAR model that only contains processes for a particular CA level. Thus, independent partial CAR models in different levels avoid overlapped processes. As a result, the CAR system could be slightly influenced with increasing CAR models, which solved (P2-A).

**(S2-B) Partial modifications when changing input**

As the solution to (P2-B), input data are processed through the hierarchy level by level, which results in different inputs and outputs of partial CAR models in different levels. This reduces the coupling between inputs and partial CAR models. Therefore, the CAR system can be partially modified for input change adaptation, which solved (P2-B).

### 1.2.4 Study's Position

Existing CAR research has achieved fairly accurate recognition for several specific retail environments and target CAs. However, flexibility, which refers to the ability to adapt to (C1) and (C2), is also important to smart retail solutions. Therefore, in this study, we study on CAR from the view of flexibility. Since a flexible CAR system is able to adapt to changes easily, it can support a wide range of retail purposes. Then, a wide range of retail purposes is supposed to deal with more types of CAs and various environments. Such circulation between CAs provided by a CAR system and retail purposes facilitates the introduction of applications in smart retail. As a result, it will greatly push the development of smart retail.

(P1) and (P2) should be solved to achieve flexibility. As the solution to (P1), we proposed (S1) a primitive-based method to realize customer behavior recognition (CBR) by the combination of primitives. To handle with (C1), changing primitives' combinations costs much less than the necessary process of rebuilding CBR models in existing CBR methods. Consequently, the proposed CBR method helps to deal with (C1) with low cost. To cope with (C2), we proposed

(S2) a hierarchy-based CAR system contains a hierarchy that divides CAs into several levels by their abstraction degrees of information. The design of partial CAR models avoids overlapped processes between CAR models and the same inputs for all CAR models, which helps the adaptation to (C2). To sum up, this study aims at achieving good flexibility to adapt to (C1) and (C2) by (S1) and (S2).

## 1.3 Dissertation Organization

This doctoral dissertation is composed of four chapters. In addition to this chapter, the remainder of this dissertation is organized as follows:

**Chapter 2.** **Primitive-Based Customer Behavior Recognition (CBR) Method**: This chapter shows the details of our proposed primitive-based CBR method. Several LSTM-based classifiers are proposed to identify primitives. To support the CB definition by combination of primitives, there is a rule to define target CBs by primitives. The evaluation results indicate that our proposed CBR method solved (P1).

**Chapter 3.** **Hierarchy-Based Customer Activity Recognition (CAR) System**: This chapter explains our proposed hierarchy-based CAR system. Since the data are processed level by level through the hierarchy, CA levels are introduced one by one in this chapter. We also implemented a prototype of the proposed system with a CAR method applied to each level. The evaluation shows that our proposed CAR system is able to solve (P2).

**Chapter 4.** **Conclusions**: Eventually, the last chapter summarizes our findings, contributions, and future direction of this study.

# 2 Primitive-Based Customer Behavior Recognition (CBR) Method

## 2.1 Introduction

### 2.1.1 Customer Behavior (CB) and Target CB Change

The word “behavior” has different meanings depending on the context. In the context of retail, we would like to define CB as the processes of a customer’s interaction with the environment during the shopping processes [37]. It involves the processes customers follow in retail stores and the reactions they have towards products or services [46]. CB provides an insight into a customer’s attitudes, needs, motivations, preferences, etc. Futhermore, CB reveals the social, economic, and cultural influence on the retail store, as well as the effect of adopted marketing strategies or management operations [42]. Therefore, CB is commonly considered to be a kind of valuable analytic material for business management [18]. In this study, CB is regarded as one of the CAs. As there are an almost infinite number of types of CBs in retail environments, generally, specific CBs are selected as recognition targets, called target CBs, based on retail purposes. Typically, customer-centric retailing demands different target CBs to analyze the customer decision-making process. Usually, the target CB changes frequently with different products or in-store layouts because of the different customer-product interactions. For instance, trying on clothes in a clothes shop, sitting on a bed in a furniture shop, picking up a bottle from a shelf, and take an ice cream out of a freezer. Accordingly, CB recognition (CBR) methods should be modified to recognize the changed target CBs. In some cases, a current target CB is required to be discriminated, e.g., in the case of “pick a product”, discriminating whether a customer is picking a product with one hand or both hands provides information regarding the customer's effort to pick the product. Therefore, a CBR method is expected to be flexible enough to address the issue of frequent changes in the target CB.

### 2.1.2 Problem Statement

As CBR is a branch of human activity recognition (HAR), current CBR methods mainly use machine learning (ML)-based models [34] due to their remarkable accuracy in HAR tasks. Nevertheless, in contrast to HAR, CBR methods also require flexibility. To recognize different CBs by changing the target CB, i.e. to change the output of the model, ML-based models require time-consuming re-collection of training data and re-training the model. Though transfer learning can be applied in some cases for faster training [1], the inevitable step of data collection is still time-consuming. This causes current methods to be inflexible when coping with changes in target CBs. Additionally, the existing methods use models trained for the target CBs that need to be recognized for the time being, and these methods are not designed to consider the change of target CBs based on business needs. Thus, current CBR methods are not suitable for target CBR tasks in retail environments.

### 2.1.3 Proposed Solution

We mainly focus on improving the flexibility of the CBR method, which is also important for CBR tasks. To cope with target changes, we propose a primitive-based method to recognize CB by combinations of primitives, each of which is the motion of an object (e.g. body, body part, and product) or the relation between two objects. Since primitives are allowed to be combined to define the pattern of various CBs, our proposed method only needs to re-define primitive combinations to adapt to target CB changes. Also, even if the number of types of primitives is small, the number of possible combinations is enormous. Thus, our method can cover a wide range of CBs with a small number of primitives.

| CB | Meaning |
|---|---|
| Passing by the Shelf [16, 29, 34] | Pass in front of the shelf without stopping |
| Turning to the Shelf [29] | Turn around and faces to the shelf |
| Viewing the Shelf [29, 36, 49] | Look at shelf without any interaction |
| Touch the Shelf [25, 29, 34–36] | Touch the shelf and take nothing outside |
| Pick up a Product [16, 25, 29, 34–36, 44] | Take a product from the shelf |
| Return a Product [16, 25, 29, 34, 36] | Put a product back to the shelf |
| Put into Basket/Cart [29] | Put a product into basket or shopping cart |
| Holding a Product [49] | Hold a product on the hand |
| Browsing a Product in a Hand [25, 36, 39, 49] | Hold and watch a product on the hand |

Table 2.1: Common CBs in existing researches

## 2.2 Related Works

### 2.2.1 Common CBs

Researchers have proposed many methods to recognize various CBs. Figuring out what kinds of CBs exist in existing works let us know the current situation of what kinds of CBs are required. Also, it allows us to find out the common features of those CBs. The common CBs are summarized in Table 2.1:

The above nine CBs usually appear in related works. Some of the CBs describe the motion of an object, while others describe the interaction between multiple objects. It is worth mentioning that no method can cover all nine common CBs, which indicates the lack of flexibility of the existing methods. Additionally, Generosi et al. [17] recognize customers' emotions from facial expressions and speech texts, which is beyond the scope of this study. Thus, this CB is excluded from Table 2.1.

### 2.2.2 Methodology of Current CBR

In retail environments, we analyze CBs to meet the demands of customer-centric retailing. As a result, CBR tasks should not only address the issues of methodology but also consider the difficulty of application and the customer's experience. Currently, although various types of sensors are used in HAR research to acquire data on human movements, almost all research on CBR uses visual data. The major reason is that visual data-based approaches can be directly applied to video acquired by surveillance cameras in the store, which makes the application of these approaches hardware-free and avoids active customer participation [18]. In addition, visual data contains much more information than most other types of sensor data.

With the input of videos, existing CBR methods mainly use the pipeline of extracting features from consecutive frames within a certain period and recognizing behavior from the sequenced features using machine-learning-based models, especially the hidden Markov model (HMM). Popa et al. [35] proposed an HMM-based model to recognize customer's buying behavior with optical flow features. Furthermore, they improved the HMM-based model by partitioning the CB into basic actions [36], which are similar to our proposed primitives. However, they determined the basic actions using optical flow features. Thus, the model is not explainable, which results in its poor flexibility when dealing with target CB changes. Merad et al. [31] applied an HMM model for hand movement analysis and an SVM model as eye-tracking descriptors for the classification of a customer's purchasing type. The specific CB classes were not given because the authors conducted CBR indirectly. Moreover, their wearable device was difficult to apply to every customer and required customers' active participation [18].

Apart from HMM models, convolutional neural networks (CNNs) are also widely used due to their excellent performance on spatial feature extraction. Singh et al. [39] used a CNN connected with a long short-term memory (LSTM) [19] model to recognize CBs, such as hand in shelf, inspect product. In their proposed method, they avoided most object occlusions by capturing video with a top-view camera. Some improved CNN-based models [34,44] have recently been proposed to detect customers and recognize basic customer-product interactions, such as picking up products, and returning products back to the shelf, etc. Liu et al. [29] employed a dynamic Bayesian network to conduct CBR of six CBs, including turning to shelf, touching, picking, and returning, based on hand movements and the orientation of the head and body. Yamamoto et al. [49] estimated CB class in a book store based on depth features from a top-view camera and pixel state analysis (PSA) features using a support vector machine (SVM).

In addition, several studies, not using an ML-based model [16, 25], implemented a complete CBR system with an RGB-D camera. Basic CBs, such as pick, and return, were recognized, based mainly on processing depth information by background subtraction. Unfortunately, since the systems were designed for specific purposes using naive methods, their flexibility was compromised.

To sum up, although the aforementioned ML-based methods achieved improvements in CBR accuracy, they share a common limitation with respect to flexibility, which is hard to adapt to changes in target CBs. The ML models cannot be reused as long as the changed CBs are substantially different from the training data. In this event, time-consuming new training data collection and model re-training are required, which implies inflexibility.

### 2.2.3 CBR vs. Human Activity Recognition (HAR)

Furthermore, we would like to discuss the similarities and differences of several HAR methods with our approach with respect to their application to CBR. Liu et al. [27] proposed an HMM-based method that divides human activity into several phases, called “motion units”, analogous to phonemes in speech recognition. Yale et al. [48] proposed interpretable high-level features based on motion units. Different activities sharing the same motion units allow the model to derive more explanatory power from human activities. Although motion units are similar to our proposed primitives, the methods encounter two issues when applied to CBR tasks, which highlight how they differ. Firstly, these methods use data from a smartphone’s acceleration sensor. The methods require the active participation of customers, e.g., downloading an app and agreeing to its terms of service, which increases their saliency to customers. Consequently, the cost increases, and the active participation creates privacy issues [18]. Secondly, despite the fairly complete categorization of human activities based on motion units, the methods do not focus on human-item interactions. Since purchase behavior can be easily detected from cashier records, recognizing non-purchase CB becomes one of the objectives of CBR. As the main component of non-purchase CBs, human-item interactions are required in CBR tasks. As an illustration, "picking up a product" and "returning a product" would be practically identical due to their similar hand motions. Rai et al. [38] divided human activities in indoor living spaces into atomic actions, analogous to the primitives in this dissertation. The use of both visual and audio data avoided users' active participation, and the training data included human-item interactions. The authors improved recognition accuracy by training the model with annotations of both atomic actions and human activities. In contrast, we concentrated on improving the method's flexibility without sacrificing too much accuracy, as flexibility is one of the important factors for CBR tasks. Mansour et al. [30] combined a faster RCNN and a deep Q network for the detection of anomalous entities or human activities in videos. Since this is a typical ML-based HAR method, it requires re-collecting training data and re-training models to adapt to the changed recognition targets, which is inflexible for CBR tasks. In conclusion, the HAR methods described require major modifications before they can be applied to CBR tasks.

## 2.3 Proposed Primitive-Based CBR Method

### 2.3.1 Overview

To achieve flexible CBR, we proposed a method to recognize CB from the video captured by in-store cameras. The proposed method recognizes CB by combination of primitives, each of which is an entity ' s motion or relation between two entities for each frame of the video.

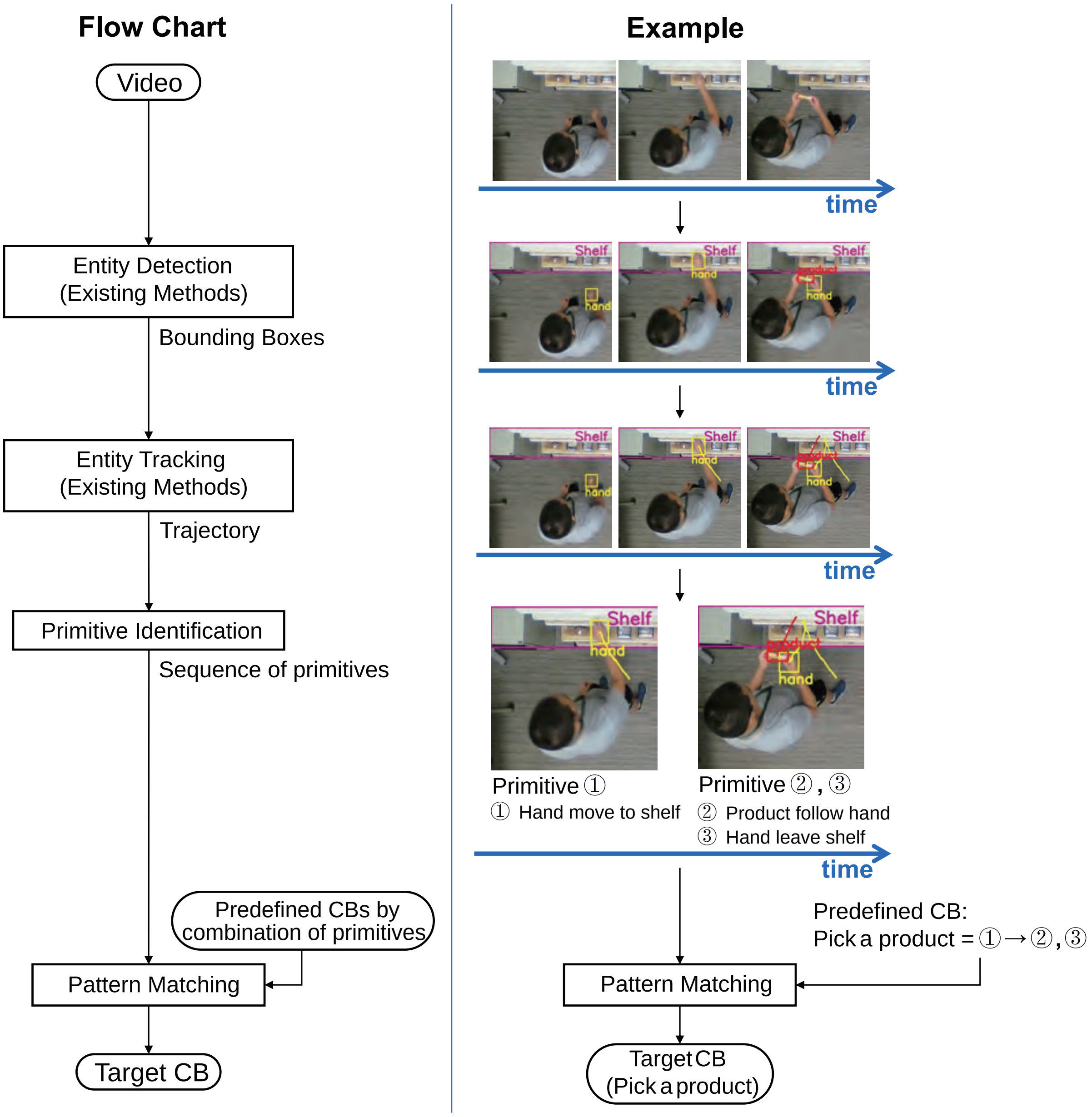


Figure 2.1: Workflow and example of our proposed method

An entity is a collective term that includes body, body part, product, or any object related to the services of a retail store. Figure 2.1 shows the workflow of our proposed method. The input is the video captured by in-store cameras. Firstly, we use existing methods to detect the entities' bounding boxes in each frame and track their moving routes, namely trajectories. The right side of Figure 2.1 is a visualization example of the outputs from each process. The purple region on the top of each frame is the product shelf. The yellow and red bounding boxes show the current location of a hand and a product in each frame, respectively. The yellow and red lines are the trajectories of hand and product. Then, we identified each frame's primitives from the trajectories. The primitives in several frames form a sequence along the time. For the example in Figure 2.1, the primitive ①, hand move to shelf, is identified at first. The yellow trajectory shows the hand's motion, which is moving to the shelf. After primitive ①, primitives ② and ③ are identified. Primitive ② reveals the relation between hand and product, which is product is following hand. And primitive ③ shows the hand movement away from the shelf. The identified primitive sequence is matched with target CBs predefined as the combinations of primitives. For example, target CB 1 "Pick a product," is defined as the primitive combination, where primitive ① happens and then ② and ③ happen concurrently. To sum up, the flowchart shows that we recognize target CBs by matching the target CB predefinition from the sequence of primitives. Therefore, we can flexibly adapt to target CB changes by re-define new target CBs by primitives' combination. This section explains our proposed method in detail, including how we design the primitives, the method for primitive identification, CB definition by primitives' combinations, and CBR by pattern matching.

### 2.3.2 Primitive Identification

In this study, CB is defined as the processes of a customer's interaction with the retail environment [37]. The definition reveals that CB is supposed to be composed of several stages that happen in a certain chronological order [8]. Consequently, we recognize the CB by matching these stages with the sequence of stages that actually happens. We refer to these stages as "primitives," each of which is the motion of an entity or the relation between two entities for each frame. Therefore, we design a primitive expression to describe these stages in CB. The expression is

$$\{subject\ verb\ object\ \}, \tag{2.1}$$

| **Element** | **Options: Explanation** |
|---|---|
| $subject$ | person, hand, shelf, product, cart/basket: entity's name |
| $verb$ | move / stay: if $subject$ is moving or not |
|  | move_to / leave / along: the moving direction of $subject$ relative to $obejct$ |
|  | with / far_from: if $subject$ is moving with $object$ or not |
|  | inside / out_of: the position of $subject$ relative to $object$ |
|  | towards / -: if $subject$ towards $object$ or not |
| $object$ | person, hand, shelf, product, cart/basket: entity's name |

Table 2.2: Elements in a primitive

where the meaning of the three elements is as follows:

- $subject$: The name of the entity that is doing something expressed by $verb$.
- $verb$: Describe the motion of $subject$ or the relation between $subject$ and $oject$.
- $object$: The name of the entity that is related to $subject$.

This expression mainly describes what an entity does, what happens to it, or the relationship between two entities. When it describes an entity's motion, $object$ can be omitted. Each element can be replaced with a specific value from the options in Table 2.2.

The value of $subject$ and $object$ are entity's name. And we mainly used the five entities in Table 2.2 so far. Regarding $verb$, in some complex situations, multiple primitives are used to describe the complex motion or relation in a frame. Therefore, we design five groups of words for $verb$. Each group following the MECE rule [32], namely mutually exclusive and collectively exhaustive. In more detail, the situations covered by every word in a group do not overlap, and all the situations related to the explanation in Table 2.2 are covered by all words in a group. Thus, if multiple primitives have the same $subject$ and $object$ in one frame, their $verb$ cannot be the value from the same group. For instance, if the identification process outputs a

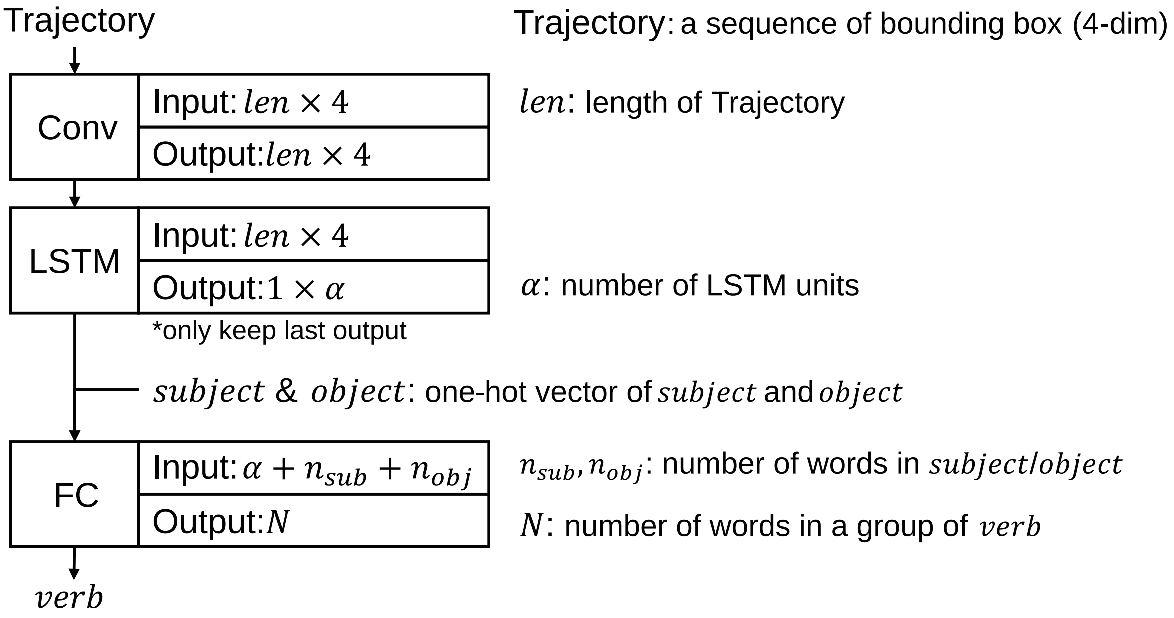

Trajectory
Conv
Input: $len \times 4$
Output: $len \times 4$
LSTM
Input: $len \times 4$
Output: $1 \times \alpha$
*only keep last output
FC
Input: $\alpha + n_{sub} + n_{obj}$
Output: $N$
$verb$
Trajectory: a sequence of bounding box (4-dim)
$len$: length of Trajectory
$\alpha$: number of LSTM units
$subject$ & $object$: one-hot vector of $subject$ and $object$
$n_{sub}, n_{obj}$: number of words in $subject/object$
$N$: number of words in a group of $verb$

Figure 2.2: LSTM-based classifier for each group of $verb$

primitive {hand move}. Then, the process should not identify another primitive {hand stay} because move and stay should be mutually exclusive since they come from the same group.

To make sure the identification outputs are mutually exclusive for the same group of $verb$ with same $subject$ and $object$, we use a model based on LSTM [19] as the classifier for each group of $verb$. Figure 2.2 shows that the classifier uses a convolutional layer and an LSTM layer to extract the temporal features from the input trajectory. The input trajectory is a sequence of bounding boxes, each of which is a four-dimension vector including the top-left and bottom-right coordinates of the entity's boundary. Then, to take the type of entity into consideration, the temporal features are concatenated with one-hot vectors of $subject$ and $object$. Then, fully connected (FC) layers are used to classify $verb$ in each group. ReLU [2] are applied as all the activation functions except the softmax activation function in the last FC layer. About the selection of $subject$ and $object$ pair, we input all the combinations of the tracked entities in a frame.

Since $subject$ and $object$ are entity's name which is the basic information of the input tra-

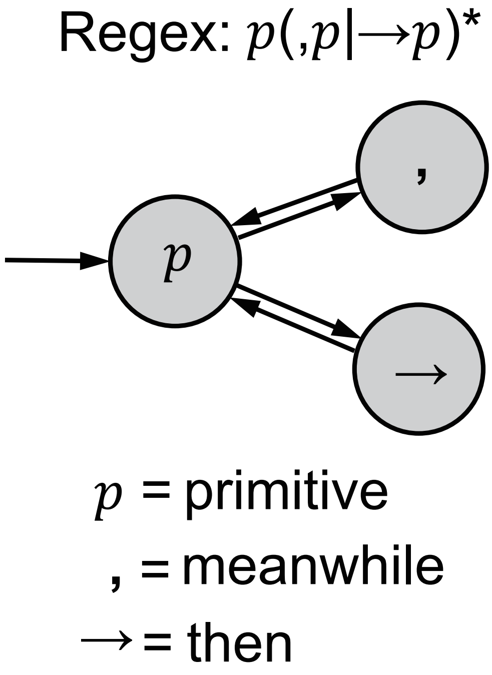


Figure 2.3: Define CB by primitives

jectory, with the LSTM-based model to identify $verb$, we can identify primitives in each frame. Consequently, primitives of consecutive frames can be merged into a sequence as the output of primitive identification process.

### 2.3.3 CB Definition by Primitives

To recognize CB from the sequence of primitives, we should define CB by the pattern of primitives. Figure 2.3 shows the way of defining CB. The regular expression is $p(,p| \rightarrow p)*$. "$p$" is primitive. "," means its adjacent primitives happen concurrently. "$\rightarrow$" refers to its previous adjacent primitive happening first, then its latter adjacent primitive happening.

For example, the CB "pick a product" can be defined as

**{hand move_to shelf}** $\rightarrow$ **{hand leave shelf}** , **{product with hand}**.

Once we replace the comma and the arrow with the word in Figure 2.3, this definition can be read as “hand move to shelf. Then, hand leave shelf. Meanwhile, product with hand.” This defines the primitive pattern for one of the target CBs. All of the target CBs are predefined in this format before running the CBR processes.

### 2.3.4 Matching CB Definition from Primitive Sequence

In the last step, we match our predefined target CBs with the sequence of primitives by Algorithm 1 and Algorithm 2 based on the BM algorithm [7].

Algorithm 1 preprocess the primitive sequence by maximum filter to filter out the noise of short duration. The input $P_{seq}$ is the sequence of primitives that comes from the primitive identification process. The algorithm uses a window to slide over the whole sequence. Primitives in the window are counted, and the most counted primitive in the window is saved into a new sequence. In case that there are multiple most counted primitives in the window, the algorithm saves the pritmive that happens earlist. The output $newP_{seq}$ is the sequence after maximum filtering. The time complexity is $O(n)$, where $n$ is length of the primitive sequence.

The input of Algorithm 2 includes the preprocessed primitive sequence $newP_{seq}$, one of the CB definitions $P_{def}$, and a parameter $timeout$ that specifies the maximum number of matching steps. The algorithm matches this CB definition $P_{def}$ in the primitive sequence. The matching starts from the end of the sequence. Only if the latest frame’s primitives matches the end of the CB definition, it is possible to succeed matching. Then, the algorithm continues matching earlier frames one by one until the unmatched frame count $counter$ exceeds the maximum value $timeout$. The output is a boolean value that represents whether the current CB definition is matched or not. The time complexity is $O(mn)$, where $m$ is the number of predefined CBs and $n$ is the length of $newP_{seq}$.

**Algorithm 1:** Preprocess primitive sequence

```
Input: List Of Primitive $P_{seq}$
Output: List Of Primitive $newP_{seq}$
$windowSize = 5$;
$newP_{seq} = []$;
for $seqIndex = 0$ to $P_{seq}.length - windowSize$ do
    /* Count primitives in the window */
    $window = []$; $count = []$;
    for $bias = 0$ to $windowSize$ do
        if $P_{seq}[seqIndex + bias] in window$ then
            $windowsIndex = window.IndexOf(P_{seq}[seqIndex + bias])$;
            $count[windowIndex] = count[windowIndex] + 1$;
        else
            $window$.Add($P_{seq}[seqIndex + bias]$);
            $count$.Add(1);
        end

    end
    /* Save the most counted primitive into the new sequence */
    $maxIndex = count.IndexOf(Max(count))$;
    $newP_{seq}$.Add($window[maxIndex]$);
end
return $newP_{seq}$;
```

**Algorithm 2:** Pattern matching of predefined CB

```
Input: List Of Primitive $newP_{seq}$, List Of Primitive $P_{def}$, Integer $timeout$
Output: Boolean $matched$
$seqIndex = newP_{seq}$.length; $defIndex = P_{def}$.length; $counter = 0$;
while $(seqIndex > 0)$ AND $(counter \leq timeout)$ do
    if $newP_{seq}[seqIndex] = P_{def}[defIndex]$ then
        $defIndex = defIndex - 1$;
        if $defIndex = 0$ then
            return True;
        end

    else
        $counter = counter + 1$;
    end
    $seqIndex = seqIndex - 1$;
end
return False;
```

## 2.4 Evaluation

Since our proposed method predefines target CBs by combinations of primitives and recognizes the target CBs by matching the CB definitions and the identified primitives' sequence, our proposed method is flexible enough to cope with the changed target CBs in different retail situations by changing the target CB definitions. To evaluate our proposed method, we used our collected laboratory dataset [45] and the public MERL dataset [39].

### 2.4.1 Dataset

**Our Laboratory CAR Dataset**

This is a dataset we collected at a public activity where 19 randomly selected participants were requested to simulate shopping in front of the shelf one-by-one. The dataset includes 19 top-view videos of 19 subjects with a resolution of 640 × 480. Each video was about 30–60s with 10 FPS and only one subject. Figure 2.4 shows the examples of the annotated CBs in the dataset. We built a laboratory retail environment and installed an RGB top-view camera to obtain an occlusion-free view. Each participant was asked to interact with the products on the shelf. Participants were required to take at least one product from the shelf. There were four products of different shapes and sizes, including a boxed juice, a deodorant spray, a stainless steel bottle, and a wet-tissue bag. The products are not visible when they are on the shelf. Our data was collected when our proposed method was demonstrated in a public activity.

Since the innovative part of our proposed method involves the receipt of trajectory coordinates as inputs, we annotated the bounding box of person, hand, and four products in each frame. Then, we used a tracker with a Kalman filter and the Hungarian algorithm [5] to obtain the object's trajectory as input. Regarding the CBs, we selected eight CBs as listed in Table 2.3. Among them, the first six CBs included common CBs that are chosen from existing methods. However, with the annotation of the six CBs (walking, viewing, browse, pick, return, touch), we found that many frames still remained without annotation. Thus, we added two CBs (select by 1 hand, select by 2 hands) to fill the frames without annotations. We used some approximate definitions for some CBs, such as "browse", because the approximate definition enabled reuse of primitives with nearly no loss of recognition accuracy.

**MERL Dataset**

The MERL shopping dataset [39] is a public dataset consisting of 106 top-view videos with a resolution of 920 × 680, each of which is about two minutes long with 30FPS. All 41 subjects

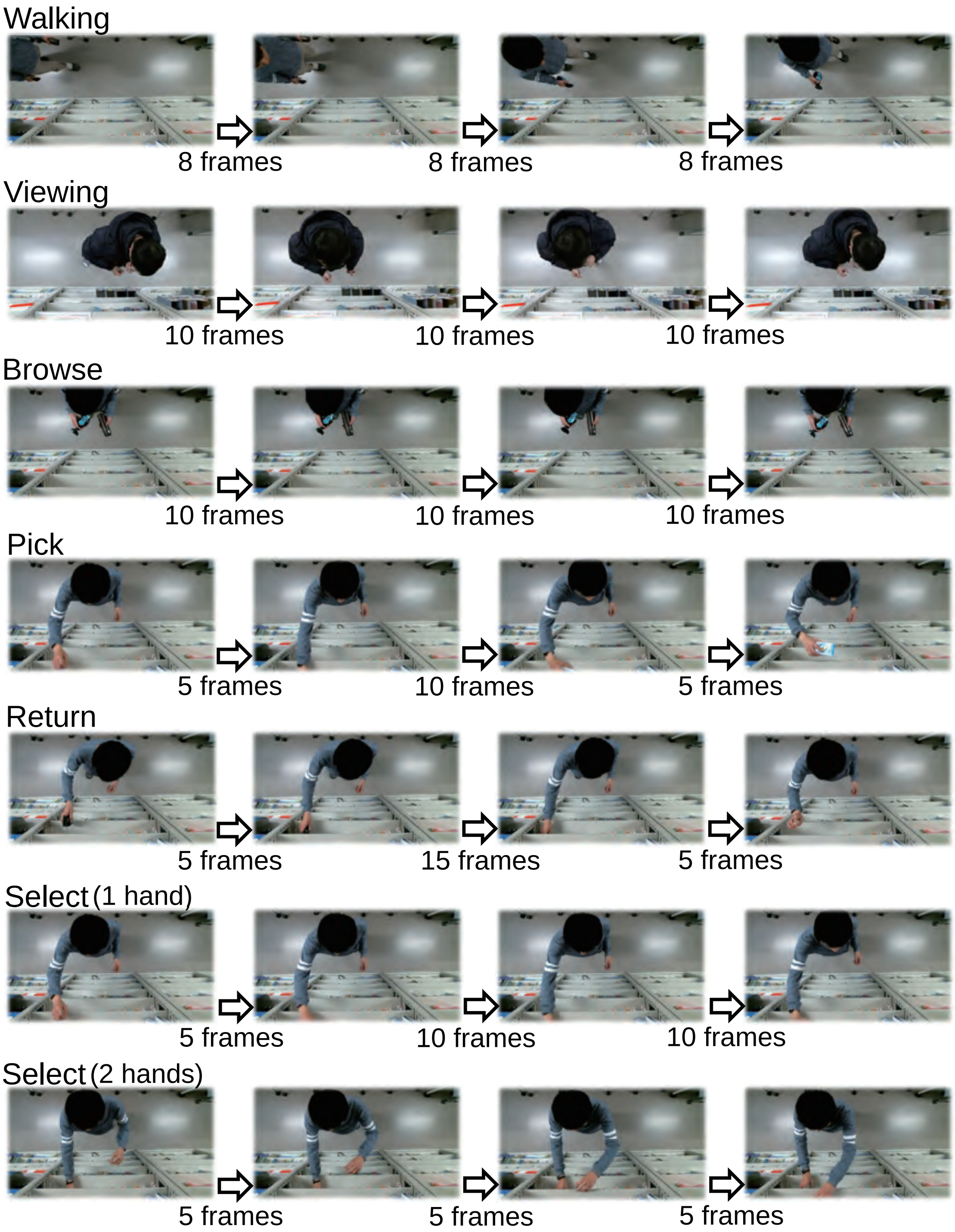


Figure 2.4: Examples of our laboratory CAR dataset

| **Target CB** | **Definition by primitives** |
|---|---|
| Viewing | {person stay},{person out_of shelf},{product with hand} |
| Browse | {person stay},{product with hand} |
| Walking | {person move},{person out_of shelf} |
| Select (1 hand) | {person stay},{hand inside shelf},{hand out_of shelf} |
| Select (2 hands) | {person stay},{hand inside shelf}, NOT {hand out_of shelf} |
| Pick | {hand move to shelf}→{hand leave shelf},{product with hand} |
| Touch | {hand move to shelf}→{hand leave shelf} |
| Return | {hand move to shelf},{product with hand} → {hand leave shelf} |

Table 2.3: CBs definitions for our Laboratory CAR dataset

were asked to shop in a retail store setting. Figure 2.5 presents some examples of the annotated CBs in the dataset. With regard to the input trajectory coordinates, we annotated the bounding box of person and hand in each frame based on the results from the pose estimation model Higher HRNet [11] pretrained on the COCO dataset [26]. We manually annotated the product's bounding box in each frame. Similar to the process for our laboratory CAR dataset, we used the same tracker with a Kalman filter and Hungarian algorithm [5] to obtain the input trajectories.

For the output CBs, we used the CB annotations in the dataset as CBs. The MERL dataset provides five CBs' annotations, and we defined them by primitives in Table 2.4. Among the five CBs, we excluded the CB "hand in shelf" from the evaluation because many ground truths were not annotated during our random checking the annotations.

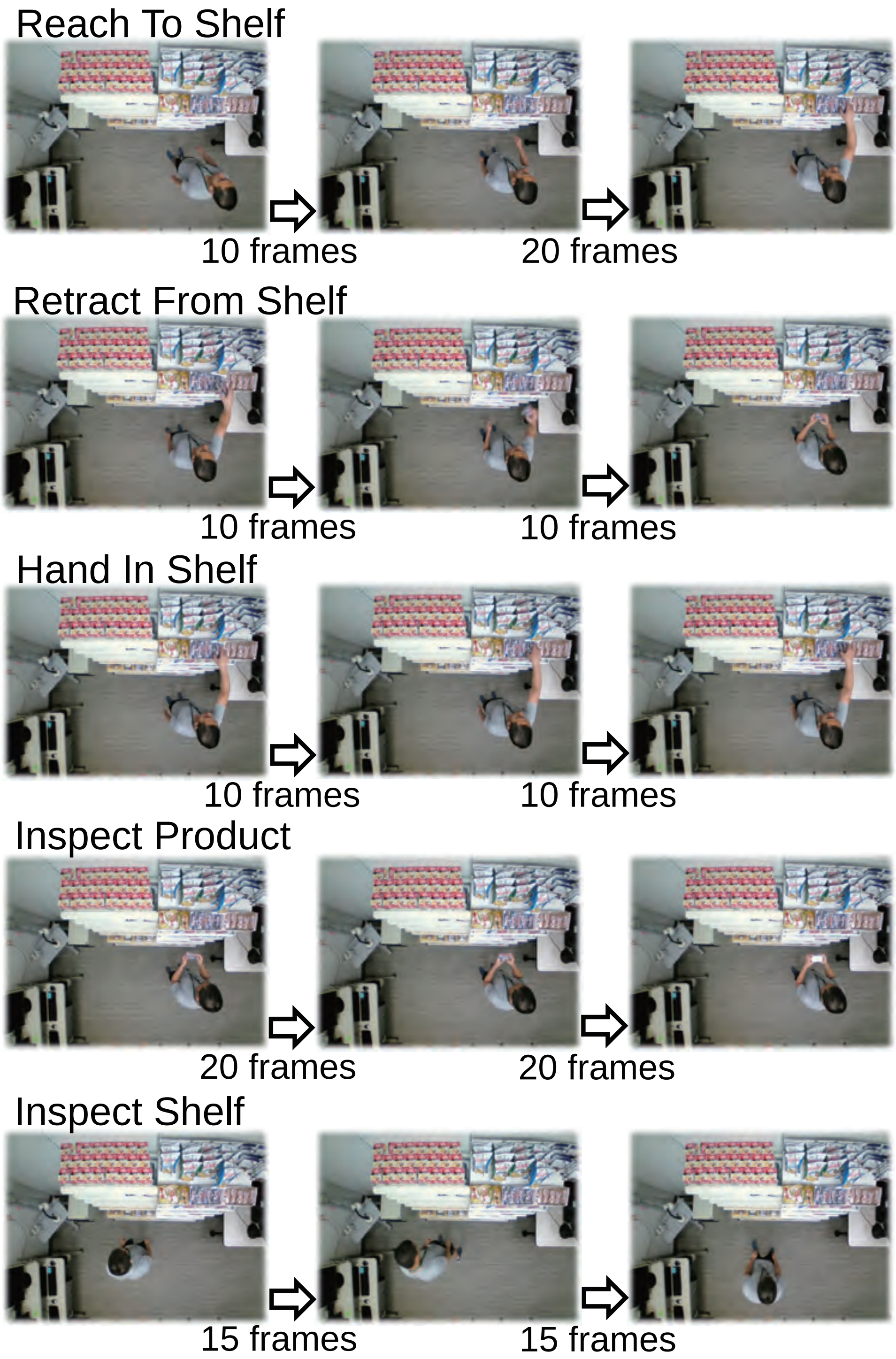


Figure 2.5: Examples of MERL dataset

| CB | Definition by primitives |
|---|---|
| Reach to shelf | $\{$hand out_of shelf$\} \rightarrow \{$hand inside shelf$\}$ |
| Retract from shelf | $\{$hand inside shelf$\} \rightarrow \{$hand out_of shelf$\}$ |
| Hand in shelf | $\{$hand inside shelf$\}$ |
| Inspect product | $\{$person stay$\}$, $\{$product with hand$\}$ |
| Inspect shelf | $\{$person stay$\}$, $\{$person out_of shelf$\}$ |

Table 2.4: CBs definitions for MERL dataset

### 2.4.2 Experiment Settings

#### Inputs and Outputs

The inputs of our method are the trajectory coordinates, which need to be obtained using object detection and a tracking model. However, wrong tracking results obtained by other models lead to wrong inputs to our method, which probably leads to wrong outputs. To eliminate the influence of different object detection models on our evaluation results, we track the annotated bounding boxes with a Kalman-filter and the Hungarian algorithm [5] to obtain a high quality of input trajectories for our method. In addition, although some tracking models can predict the trajectories of occluded objects, the occluded trajectories are not annotated in both datasets. About the output CB annotations, we annotated the target CB in each frame for our laboratory CAR dataset. As the MERL dataset is public, we used its original CB annotations.

#### Training of Classifiers in Primitive Identification

To train the classifiers in primitive identification, we annotated the primitives in our CAR dataset. The coordinates of trajectories are normalized by the resolution. 80% of the annotations are used as training sets, and the rest are testing sets. Since there are five groups in $verb$, we trained five classifiers for each group. The hyperparameters, including $len$ and $\alpha$ in Figure 2.2,

| ***Verb* Group** | ***len*** | $\alpha$ | **Conv kernel size** | **Learning rate** $(\times 10^3)$ | **Batch size** |
|---|---|---|---|---|---|
| move / stay | 13 | 62 | 1 | 6.08 | 175 |
| move_to / leave / along | 11 | 57 | 1 | 4.56 | 143 |
| with / far_from | 16 | 55 | 1 | 1.14 | 202 |
| inside / out of | 7 | 38 | 1 | 0.55 | 281 |
| towards / - | 20 | 52 | 1 | 2.16 | 265 |

Table 2.5: Classifier training results

learning rate, and batch size, are optimized by Optuna [4]. The optimization and training results are shown in Table 2.5.

### Evaluation Metrics

Since the objective of our proposal is flexibility, we compare the modifications of our method and existing ML-based methods during the adaptation to the change of target CBs of two datasets mainly on the time and number of modifications. Also, we calculate the f1-score of the recognition results of our proposed methods on two datasets as the accuracy metric. By observing the adaptation comparison and the accuracy of our method on different datasets, we could infer our method's flexibility to some degree.

### Devices

For the experiments on both datasets, we implemented our method on the same Windows 11 device with RAM of 16 GB. The CPU was an Intel i7-12700K (3.6 GHz). The GPU was an NVIDIA GeForce RTX 3060 Ti (8 GB). The program was written in Python 3.9. The ML framework was PyTorch 1.12. The used third-party libraries include numpy 1.22 and scipy 1.8.

| Laboratory CAR dataset | → | MERL dataset |
|---:|:---:|:---|
| Viewing | = | inspect shelf |
| Browse | = | inspect product |
| walking | → | - |
| select(1 hand) | → | - |
| select(2 hands) | → | - |
| pick | → | - |
| touch | → | - |
| return | → | - |
| - | → | reach to shelf |
| - | → | retract from shelf |

Table 2.6: Target CB changes

### Experiment

In the experiment, we first applied our proposed method to the target CBs from the laboratory CAR dataset. Then, we change the target CBs to be from the MERL dataset. Table 2.6 shows the changes in target CBs from different datasets. About the existing ML-based methods, because of the huge amount of works to reproduce common existing models, we estimate the adaptation cost of existing methods by only the re-annotation time and number of parameters without building and training existing models. Besides, since accuracy is not the focus of our research, we only provide the accuracy results of our proposed methods instead of comparing our proposed method with existing methods.

| Method | Modifications | Reference Time |
|---|---|---|
| Our Proposed Method | Define new target CBs by primitives | 2-3 man hours |
| Existing ML-based Method | Collect training data | - |
| | Annotate training data | 176 man-hours |
| | Train model | - |

Table 2.7: Adaptation cost comparison

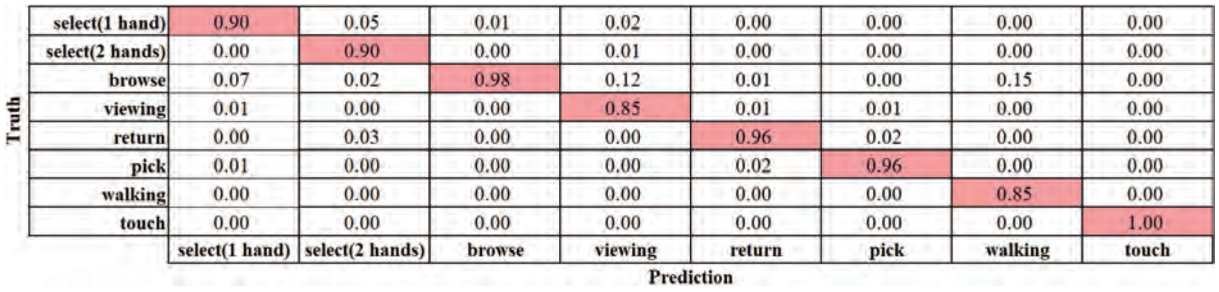
### 2.4.3 Adaptation Cost Comparison

Table 2.7 compares the cost during the adaptation to changed target CBs. For our proposed methods, we defined the new target CBs by primitives, which took us about 2–3 man hours. For existing ML-based methods, one of the necessary modifications should be to annotate new target CBs in the dataset, which took us about 176 man hours. Also, there are usually thousands of parameters for a deep-learning network, which is much more than the number of parameters in our proposed methods. To sum up, the comparison indicates that our proposed method is more flexible because of the fewer modifications and quicker adaptation to target CB changes.

### 2.4.4 Accuracy Results

**Results on Our Laboratory CAR Dataset**

Figure 2.6 shows the confusion matrix of our laboratory CAR dataset. Each value is the ratio of its column. Figure 2.7 shows the f1-score and some statistics for our laboratory CAR dataset. The total average f1-score of our method is 92.04%. The f1-score for each CB is different. Most target CBs are recognized but the duration is different from the annotations, which reduces the f1-score. In terms of the low precision of “viewing,” the confusion matrix reveals the reason with 84.9% precision. Some “viewing” frames are recognized as “select.” The ambiguous boundary during the transition from “viewing” to “select” causes the wrong recognition. Also,



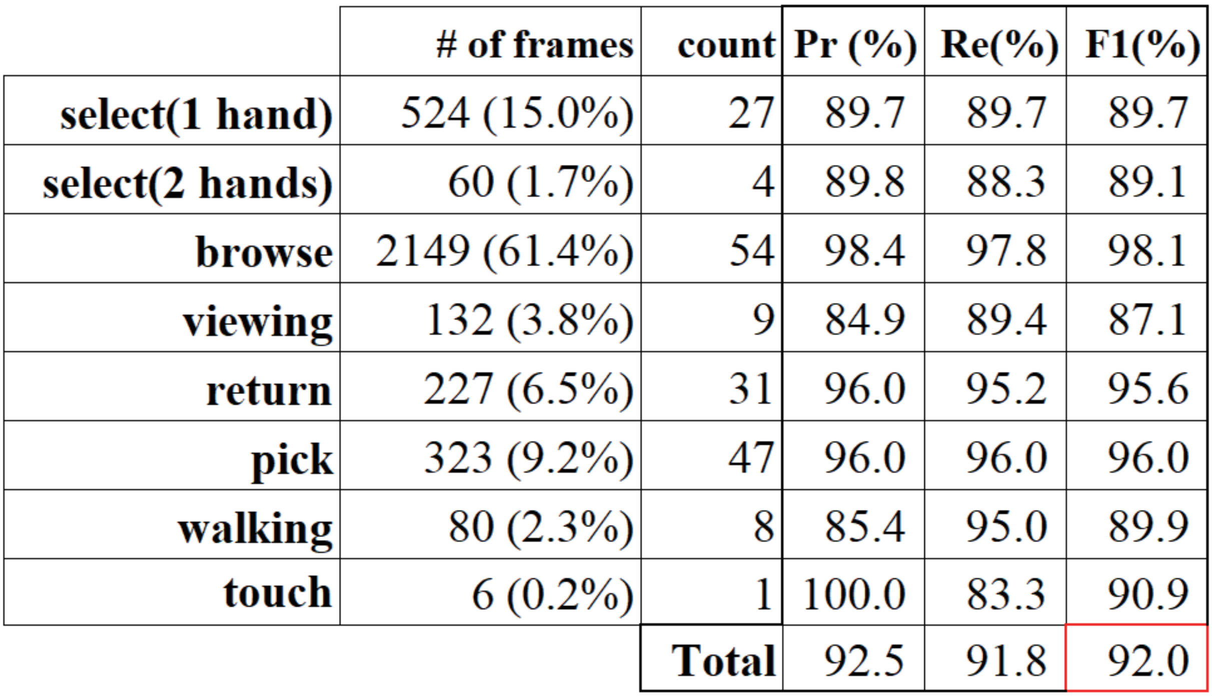
| Truth | select(1 hand) | select(2 hands) | browse | viewing | return | pick | walking | touch |
|---|---|---|---|---|---|---|---|---|
| select(1 hand) | 0.90 | 0.05 | 0.01 | 0.02 | 0.00 | 0.00 | 0.00 | 0.00 |
| select(2 hands) | 0.00 | 0.90 | 0.00 | 0.01 | 0.00 | 0.00 | 0.00 | 0.00 |
| browse | 0.07 | 0.02 | 0.98 | 0.12 | 0.01 | 0.00 | 0.15 | 0.00 |
| viewing | 0.01 | 0.00 | 0.00 | 0.85 | 0.01 | 0.01 | 0.00 | 0.00 |
| return | 0.00 | 0.03 | 0.00 | 0.00 | 0.96 | 0.02 | 0.00 | 0.00 |
| pick | 0.01 | 0.00 | 0.00 | 0.00 | 0.02 | 0.96 | 0.00 | 0.00 |
| walking | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.85 | 0.00 |
| touch | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 1.00 |

Figure 2.6: Confusion matrix of our laboratory CAR dataset

the difference between approximate CB definition and annotation of target CB “viewing” leads to the wrong outputs. As our proposed method cannot recognize the target’s orientation or track the target’s eyes currently, our CB definition approximately defines “viewing” as stay static out of the shelf, while the annotation of “viewing” means the target is standing still and looking at the shelf. This is also the reason to the low recall of “viewing.” Besides, products are usually occluded in the “pick” and “return” frames, which causes the wrong recognition output, “viewing.”

With respect to “walking”, some of its frames are recognized as “browse”. When the target is walking while holding a product, it is difficult to determine the ambiguous boundary between “browse” and “walking”. The CB definition in Table 2.3 recognizes them by distinguishing whether the target is moving while holding a product. “Browse” refers to holding a product while remaining static. Some moving frames are detected as staying static, which led to the wrong recognition. This also applied to the low precision of “walking.”

In the case of “touch”, there is only one case in the dataset. It was defined as a customer putting their hand inside the shelf but taking nothing out of it. Some wrong recognition of “pick” results in the low recall occurred because the picked object was occluded. In addition, Figure 2.6 shows that most video frames are “browse” and occur more frequently than any other CBs. Thus, we consider discriminating within “browse” to make the distribution of CBs more uniform.



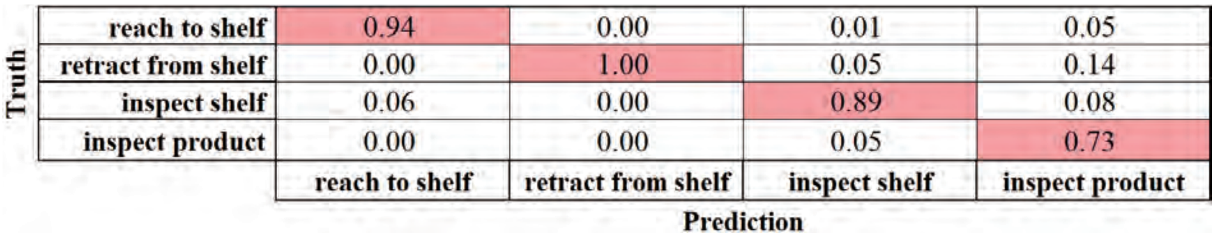
| | # of frames | count | Pr (%) | Re(%) | F1(%) |
|---|---|---|---|---|---|
| select(1 hand) | 524 (15.0%) | 27 | 89.7 | 89.7 | 89.7 |
| select(2 hands) | 60 (1.7%) | 4 | 89.8 | 88.3 | 89.1 |
| browse | 2149 (61.4%) | 54 | 98.4 | 97.8 | 98.1 |
| viewing | 132 (3.8%) | 9 | 84.9 | 89.4 | 87.1 |
| return | 227 (6.5%) | 31 | 96.0 | 95.2 | 95.6 |
| pick | 323 (9.2%) | 47 | 96.0 | 96.0 | 96.0 |
| walking | 80 (2.3%) | 8 | 85.4 | 95.0 | 89.9 |
| touch | 6 (0.2%) | 1 | 100.0 | 83.3 | 90.9 |
| | | Total | 92.5 | 91.8 | 92.0 |

Figure 2.7: Accuracy results of our laboratory CAR dataset (Pr = Precision, Re = Recall, F1= f1-score)

Among the above results, some CBs with a low f1-score are anticipated to be improved by changing the CB definitions into more accurate definitions. Also, to evaluate our proposed method's ability to adapt to changes of target CB, we discriminated the CB "select." We predefined different primitive patterns to discriminate "select" according to whether one hand or both hands were used. This indicates that our proposed method is able to deal with CB discrimination, which is also a kind of change of target CB, to some extent.

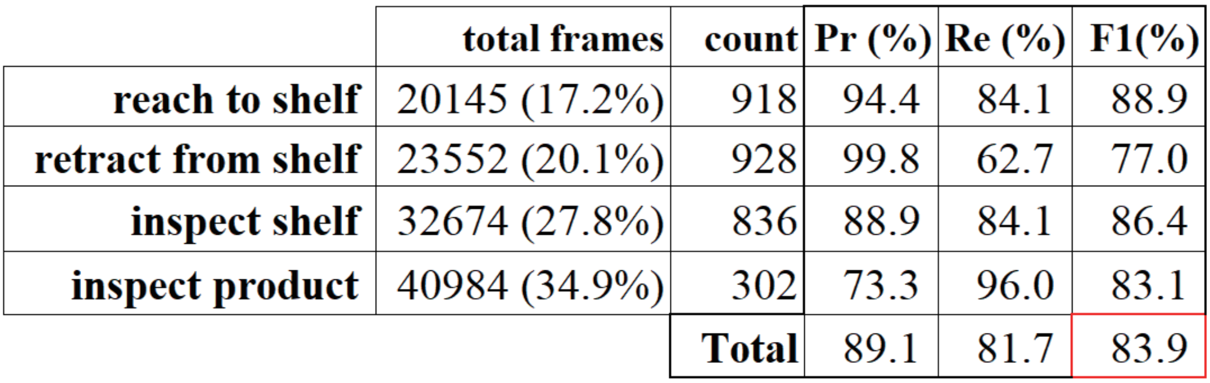
### Results on MERL dataset

Figure 2.8 shows the confusion matrix of the MERL dataset. Each value is the ratio of its column. Figure 2.9 shows the f1-score and statistics for the MERL dataset. Our proposed method achieves the average f1-score of 83.86% by only changing CB definitions. Among the four target CBs, our method  achieve only about 62.7% recall for "retract from shelf." We found that



| Truth | reach to shelf | retract from shelf | inspect shelf | inspect product |
|---|---|---|---|---|
| reach to shelf | 0.94 | 0.00 | 0.01 | 0.05 |
| retract from shelf | 0.00 | 1.00 | 0.05 | 0.14 |
| inspect shelf | 0.06 | 0.00 | 0.89 | 0.08 |
| inspect product | 0.00 | 0.00 | 0.05 | 0.73 |
| | Prediction | | | |

Figure 2.8: Confusion matrix of MERL dataset



| | total frames | count | Pr (%) | Re (%) | F1(%) |
|---|---|---|---|---|---|
| reach to shelf | 20145 (17.2%) | 918 | 94.4 | 84.1 | 88.9 |
| retract from shelf | 23552 (20.1%) | 928 | 99.8 | 62.7 | 77.0 |
| inspect shelf | 32674 (27.8%) | 836 | 88.9 | 84.1 | 86.4 |
| inspect product | 40984 (34.9%) | 302 | 73.3 | 96.0 | 83.1 |
| | | Total | 89.1 | 81.7 | 83.9 |

Figure 2.9: Accuracy results of MERL dataset (Pr = Precision, Re = Recall, F1= f1-score)

this was caused by the different boundary between CB definition and annotation. Specifically, there was a difference between our definition of “retract from shelf” and the definition in the MERL dataset. Our method starts to recognize “retract from shelf” from the frame in which the hand was already moving. The MERL dataset defines the start of “retract from shelf” as when one intends to retract from shelf, when one’s hand has not yet moved. Thus, our recognition results always differed from the annotations by a few frames. We consider our method to have been successful in recognizing every “retract from shelf” CB with a few frames’ difference. This implies that we could improve our method by recognizing intention in our future research.

## 2.5 Summary

In most cases, smart retail solutions need to be able to recognize a wide range of CBs from in-store videos. The CBs that are selected as recognition targets usually change depending on various retail situations or purposes. The objective of this chapter is to realize a flexible CBR method that can adapt to various changes of target CBs. We proposed a primitive-based CBR method. It recognizes CB by combination of primitives, each of which is an entity's motion or two enties' relation. The proposed method allows us to define a wide range of target CBs by the combination of primitives. Therefore, to adapt to the changed target CBs, we only need to define the changed target CBs by primitives. Compared to the high cost adaptation steps of existing ML-based methods, we achieved flexible CBR to adapt to the target CBs change in smart retail, which solved (P1) Hard to Adapt to Target Change.

Nevertheless, the time complexity of the pattern matching algorithm reduces the running performance a lot when there are lots of entities in one frame. Therefore, our proposed method is only available for one-person scenes in retail stores. Also, currently a few cases requires the orientation information in primitives, which could be the future direction of our work.

# 3 Hierarchy-Based Customer Activity Recognition (CAR) System

## 3.1 Introduction

### 3.1.1 CAR System and Environment Changes

CA refers to various customer's situations in store spaces. CA provides valuable information for store management (e.g. layout optimization and shoplifting prevention), marketing planning (e.g. supply control and product development) [18, 31], etc. Therefore, it is essential to build a system that includes various CAR methods to provide various CAs. The system is called CAR system in this study. Though the output CAs should be accurate enough to be reliable, the system should also be flexible to adapt to the environment changes in smart retail. The environment refers to in-store implementations for retail service or management, including product types, surveillance cameras, and store layout. The environment usually changes for needs like different store types, festival promotion, inventory supply, security and so on. For instance, changes in inventory supply may lead to changes in store layout or product types. Therefore, we should modify the CAR system to adapt to the new layout or new products.

### 3.1.2 Problem Statement

Most of the existing CAR systems use machine learning-based End-to-End (E2E) models due to their remarkable accuracy in CAR tasks [34]. Usually, such E2E models in CAR systems are constructed as complete CAR models, each of which is trained to recognize particular types of

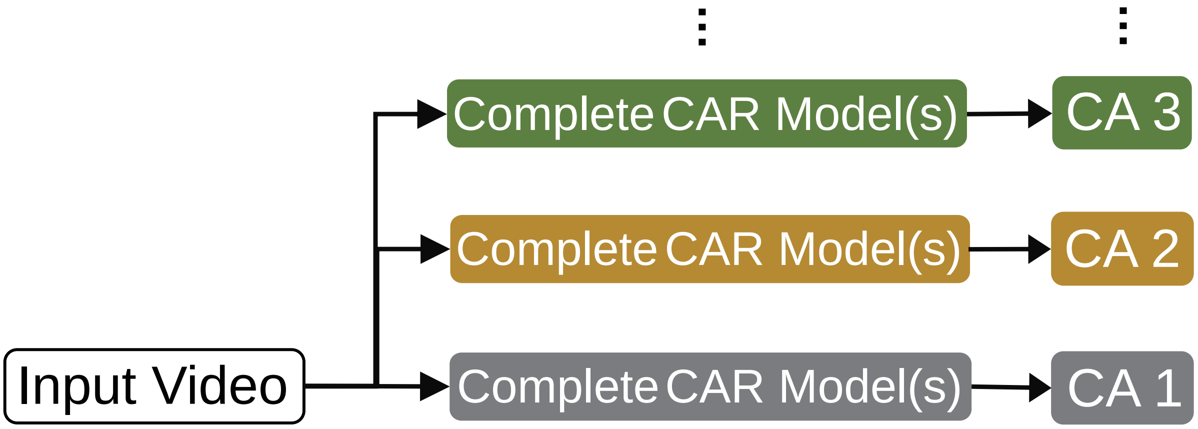


Figure 3.1: Structure of existing CAR system

CA from input videos. Consequently, as shown in Figure 3.1, existing CAR system runs several E2E complete CAR models in parallel with the same input video. When additional types of CA is required, an additional complete CAR model, whose input is the same video as the other complete models, is added to the CAR system. However, this system structure leads to the problems when adapting to the changes of environment changes.

- **Reducing performance when increasing CAR models**: CAs can be classified into levels by their abstraction degree of information, thus high level CA should be the abstraction of several low level CAs. The processes of complete CAR models for high level CA and low level CA must have overlapped parts. When environment changes need to add CAR model(s), e.g. introducing new products leads to additional CAR model(s) for product detection, the overlapped processes lead to reducing performance of the CAR system with increasing number of CAR models. This makes the CAR system hard to adapt to environment changes.

- **Modifications for all CAR models when changing inputs**: Since the complete CAR models share the same inputs, inputs and CAR models are highly coupled. If inputs change due to environment changes, such as changing the camera to a new one with a

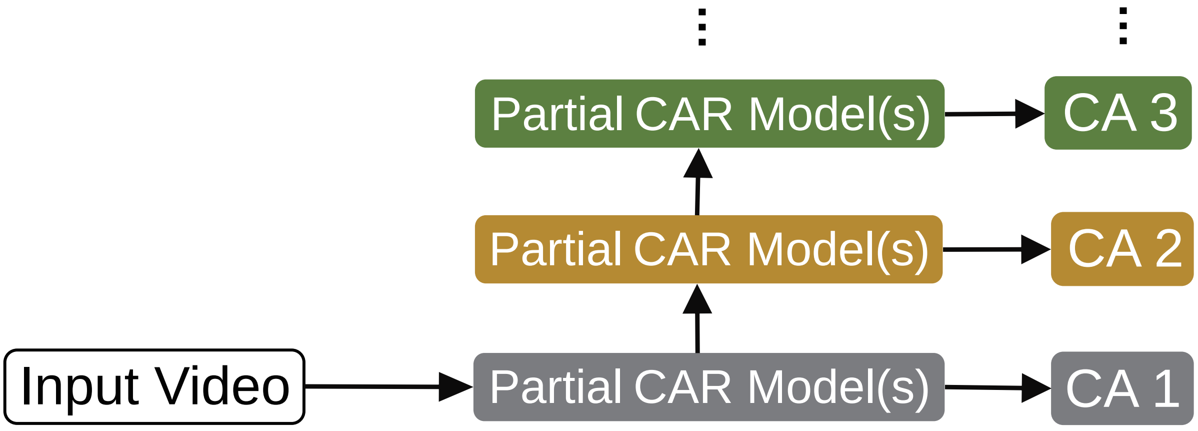


Figure 3.2: Structure of proposed CAR system

different resolution or view, modifications for all CAR models are inevitable. This also leads to difficulty in adapting to environment change.

### 3.1.3 Proposed Solution

In this chapter, we propose a novel CAR system based on a hierarchy that organizes CA types into different levels of abstraction from lowest to highest. As shown in Figure 3.2, the proposed system is composed of multiple partial CAR models, each of which corresponds to the recognition process of a certain CA level and performs the CAR process on the lower level output. The proposed CAR system has the following properties to solve the problems mentioned above:

- **Slight influence when increase CAR models**: The complete CAR models are divided into partial CAR models corresponding to the CA levels. Each level includes a partial CAR model that only contains processes for a particular CA level. Thus, independent partial CAR models in different levels avoid overlapped processes. As a result, the CAR system could be slightly influenced with increasing CAR models.

| **CA Level** | **CA** | **Related Works** |
|---|---|---|
| Spatial Features | Entity's Location | [9,10,12,14–16,22–25,28,29,34,40,41,49,50,52] |
| Temporal Features | Optical Flow | [35,36] |
| | Trajectory | [16,25,28,29,33,35,41,49,51] |
| Behavior | CB | [16,25,29,34–36,39,44,49] |

Table 3.1: CA levels in the existing CAR systems/methods

- **Partial modifications when changing input**: Since the input of each partial CAR model is the output from its previous level, different inputs lead to loose coupling between inputs and CAR models. As a result, when the environment change makes the changes in input videos, the proposed CAR system only needs to partially modified the CAR models in CA levels related to the input videos.

## 3.2 Related Works

### 3.2.1 CA in Existing Research

Up until now, existing researches has provided lots of ways to get various types of CA. As the objective is a CAR system that is flexible enough to provide various levels of CA during the environment changes, we summarize the CA levels in existing CAR systems/methods as shown in Table 3.1.

#### Spatial Features

Extracting spatial features from each frame, like getting an entity's location, is usually the first step of many CAR approaches because they need the spatial features as a kind of basic information for further processing. Among those CAR methods, most researches focus on human detection. Different sensors are applied to detect human [18], in particular, WiFi RSS [10, 15, 52], RFID [40], GPS [14], Bluetooth Beacons [9, 12, 22], RGB camera [17, 23, 24, 29, 39, 50], RGB-Depth camera [16, 25, 28, 34, 41, 49]. Due to its descriptive nature, visual data is the preferred input for human detection. Nevertheless, few researches [41] combine multiple sensors. Conventional methods detect humans in data extracted from RGB images by using Histogram of Oriented Gradients (HOG) [23, 50]. However, more recently, machine-learning tools for human detection, especially using Convolutional Neural Network (CNN) [17, 29, 39] were developed because of their excellent performance concerning the detection speed and accuracy in a wide range of environments.

Compared to RGB-based-only methods, the combination of top-view and RGB-depth cameras provides more straightforward solutions for human detection [3]. The top-view image offers an occlusion-free view as the occlusion rarely occurs in the top-view direction, which takes images from directly above. It also preserves privacy since faces are usually not exposed to the camera. As the depth values change significantly in the human region, the legacy human detection method [16, 25, 49] is to run background subtraction on the top-view RGB-D images. Another method [28] uses self-designed features transformed from the original RGB-D pixels to detect humans. Additionally, though most methods detect the whole region of the human body, some approaches [17, 29, 31, 34] also focus on some specific body parts, such as the head and hand. Additionally, except for human detection, Merad et al. estimate behavior by detecting products and hand gestures from the images of a wearable device. Also, Zhao et al. [51] proposed a combined neural network-based model to detect products from RGB images. Finally, Paolanti et al. [33] detect humans by locating shopping carts and baskets with Ultra-WideBand (UWB) Technology.

### Temporal Features

Temporal features are abstracted from spatial features, including pixel-level optical flow or entity's bounding box trajectory. Similar to the detection tasks, more efforts have been devoted to processing visual data for tracking tasks. The tracking target tends to be humans rather than the other entities [33, 51]. Some methods track the other entities for a further purpose, such as behavior recognition. Therefore, the entity's coordinates in each frame are unnecessary. They track pixels, for instance, optical flow [35, 36]. Instead of tracking pixels, most methods track the entity's coordinates. Kalman filter [28, 49] and particle filter [29] are common solutions to the coordinate tracking tasks. Some approaches also combine both filters [41] to achieve better tracking results. In some cases of a reliable detector, the Intersection over Union (IoU) tracker is also applied [16, 25].

### Behavior

Behavior mainly refers to CB, which is a kind of combined spatial and temporal feature. Thus, CB is abstracted from spatial and temporal features. CBR is always a challenging topic [21] due to the methodology involved in features such as the inter-, intra-class variations, cluttered background, or changes of the camera views. In the context of retail, existing methods mainly share the pattern of extracting the spatial features from consecutive frames and recognizing behavior from the sequenced spatial features using models, especially the Hidden Markov Model (HMM). Popa et al. [35, 36] proposed an HMM-based model to recognize customer behavior with optical flow features. Singh et al. [39] use CNN to extract spatial features of each video frame and recognize behavior by Long-short Term Memory (LSTM) [19] with the spatial features. Compared to the pixel level features, some approaches utilize HMM [31], Dynamic Bayesian Network [29], and Support Vector Machine (SVM) [49] to analyze coordinate trajectories for behavior recognition. In addition, due to the property of some customer behavior, some methods designed self-defined rule-based models [16, 25, 34] to recognize behavior.

Common target CBs in existing CBR methods mainly includes customer-product interactions, such as “pick a product,” “return a product,” “holding a product,” and indirect behavior like “pass by,” “viewing the product shelf.” Additionally, some CBs [49] are excluded because of their misleading definition, and some are merged due to similar definitions. In some particular cases, Popa et al. [36] recognize customer’s interactions with clothes in a cloth store. Despite the variety of recognition results, few researchers comprehensively categorize customer behavior. Popa et al. [36] extract spatial and temporal features with Histograms of Optical Flow (HOF). In this research, the behavior is divided by the level of granularity of HOF. Besides, Generosi et al. [17] estimate customer’s emotion, which is also a type of customer behavior, with facial expressions and speech text. However, this feature is out of the scope of this dissertation as it might breach the privacy of customers.

### 3.2.2 Structure of Existing CAR System

Existing CAR researches tend to focus on a particular CAR method that is specialized for one retail purpose. Therefore, only a few researches [18, 25, 34, 36] provide a view of the common structure of a CAR system. According to the structure of few existing CAR systems and the low flexibility of existing CAR methods, we summarize the structure of a common CAR system as shown in Figure 3.1. A common CAR system runs several ML-based complete CAR models in parallel with the same input video. As mentioned above, the structure mainly causes two problems when adapting to environment changes. One of them is that when the environment change requires the implementation of additional CAR models, the overlapped processes between complete CAR models lead to reducing performance of the CAR system. Another one problem is that when the environment change need to change the input video, because of the sharing the same input video, all complete CAR models in the CAR system must be modified to adapt to the new input. The two problems make existing CAR systems inflexible to adapt to environment changes, namely (P2).

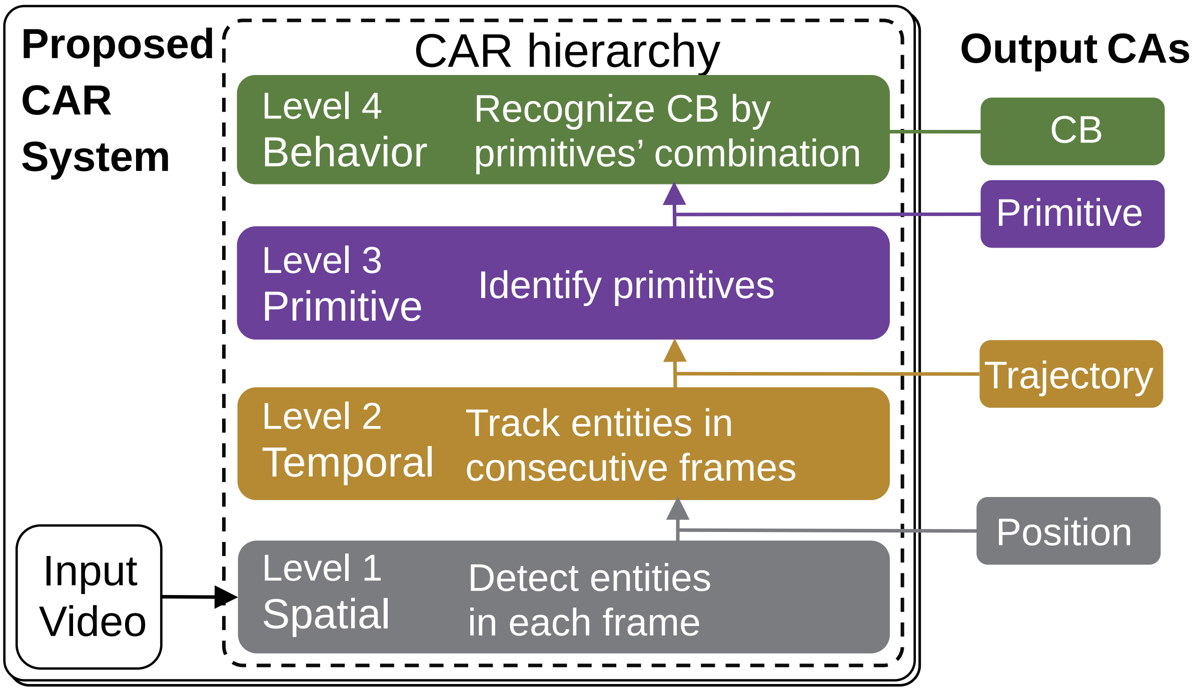


Figure 3.3: Proposed hierarchy-Based CAR system

## 3.3 Proposed CAR System

This section gives an overview of the proposed hierarchy-based system and explains the details of every hierarchy level. The structure of the system is depicted in Figure 3.3 where the video data is the input for the hierarchy. The hierarchy comprises four levels: Spatial, Temporal, Primitives, and Behavior. The job of each level is to abstract the information from its lower level. Therefore, the CAR methods in each level can be replaced by any methods as long as it can finish this level's job. Consequently, the CAR methods in any level are not limited to a particular methods, better methods can be applied to adapt to different retail scenes.

### 3.3.1 Level 1: Spatial

As the name implies, this level conducts a spatial detection process by the sensor data. Generally, all sensors have a frequency of collecting data, such as the Frame Per Second (FPS) for cameras, the sampling rate for sensors that transmit electromagnetic signals. Level Spatial extracts spatial information from each sampled data. Since visual data is mainly utilized by most detection methods, the input of this level is set as video captured by in-store cameras. Commonly, this level detects entities' bounding boxes in each camera frame.

### 3.3.2 Level 2: Temporal

This level provides the temporal features from the abstraction of spatial features in consecutive frames. The temporal features contain pixel-level features, such as optical flow, and entity-level features, such as trajectory, namely moving route. Therefore, the job of this level is the extraction of pixel-level features from the change of pixels, and entity-level features from the bounding boxes in consecutive frames. In one word, Level Temporal conducts tracking jobs with the outputs from Level Spatial.

### 3.3.3 Level 3: Primitives

Primitive is the same as that in Chapter 2. It represents an entity's motion or two entities' relation in each frame. Thus, this level's job is primitive identification. When the primitives are merged along the time, the output forms a sequence of primitives.

### 3.3.4 Level 4: Behavior

As the name implies, this level of the hierarchy performs customer behavior recognition. Since its previous level outputs a sequence of primitives, this level matching the predefined target CBs from the sequence of primitives.

### 3.3.5 Hierarchy-Based CAR System

The CAR methods are partial CAR models that only output CA of a particular level with the inputs from its previous level. Therefore, the CAR methods form a hierarchy that not only runs CAR processes level by level but also categorizes CAs into levels. The system achieves flexibility in response to environment changes as below.

- **Slight Influence on System's Performance**: The use of partial CAR models avoids overlapped processes between CAR models. As a result, when the environment change needs to add CAR models, adding a new level in the case of our proposed CAR system, it has slight influence on the system's performance compared to existing CAR systems.
- **Partially modify CAR models in the system**: When the environment change cause the change of inputs, because of different input for CAR models, we only need to partially modified CAR model(s) related to the inputs instead of modification of the entire CAR system in case of existing CAR systems. Since the CAR models in our proposed CAR have different inputs, it indicates that our CAR system can handle the environment changes that are not limited to the camera change. For instance, to cope with different customer's habits for the same CB, we only need to modify the methods in level Primitive by adding primitives to cover the new customer habits. Therefore, different inputs of the four levels imply that the four levels can handle corresponding environment changes independently.

## 3.4 Performance Evaluation

This section evaluates the running performance and adaptation cost of a CAR system when applied to different retail environments.

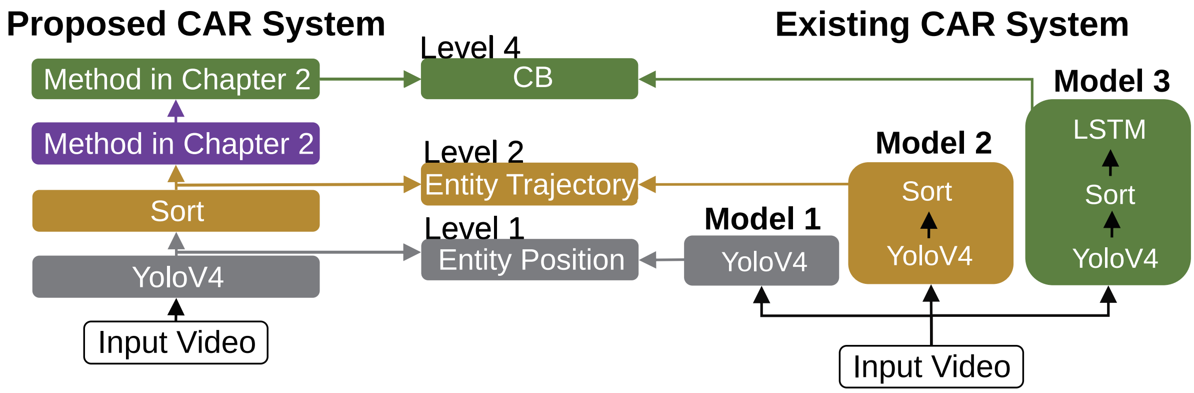


Figure 3.4: Proposed hierarchy-Based CAR system

### 3.4.1 Implementation

Figure 3.4 shows the implementation of our proposed hierarchy-based CAR system and an existing E2E ML model-based CAR system. The video is used as the original input for both systems. For our hierarchy-based system, YOLOv4 [6] and Sort [5] are respectively implemented into level Spatial and Temporal for object detection and tracking. In level Primitive and Behavior, we use our proposed methods in Chapter 2. For the E2E ML model-based system, we build three models to simulate complete CAR models. Though they look like cascading models, each of them is considered as a complete CAR model that has the input of video and the output for only one level of CA. Three models are implemented for level Spatial, Temporal and Behavior. Also, both systems do not output primitives because there are no existing methods for primitive identification. It should be mentioned that though existing CAR system outputs different levels' CAs simultaneously in Figure 3.4. We actually run Model 1,2,3 one by one to output three levels' CAs. In fact, the simultaneous outputs are not necessary in applications because different levels' CAs are used to support different retail purposes. Most of the retail purposes do not require real-time CAs.

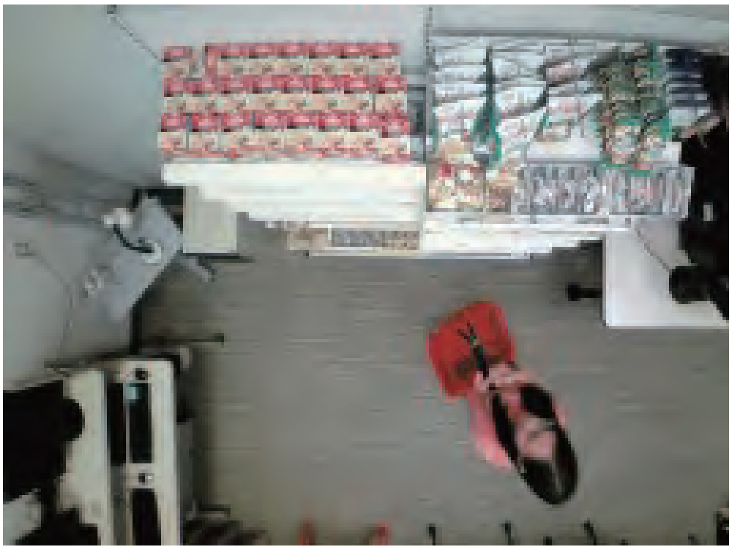


Figure 3.5: Example of MERL dataset

### 3.4.2 Dataset

Two datasets are utilized in the experiments, one is the MERL shopping dataset [39] that consists of 106 videos. Each video is about 2 minutes long with a resolution of 920x680 and an FPS of 30. A fixed RGB top-view camera is installed to observe people shopping in a retail environment. Figure 3.5 shows an example of MERL dataset.

Another dataset is our collected laboratory CAR dataset. It comprises 19 half-minute videos with a resolution of 640x480 and an FPS of 10. We build a laboratory retail environment and install an RGB top-view camera to get an occlusion-free view. Figure 3.6 shows its example frame. Videos are collected at a public activity, where 19 random participants are requested to simulate shopping by standing in front of the shelf one by one. Both datasets have the annotation of entities' bounding boxes and CB for each frame. CB annotations include walking, viewing, browse, pick, return, touch, select (1 hand), and select (2 hands).

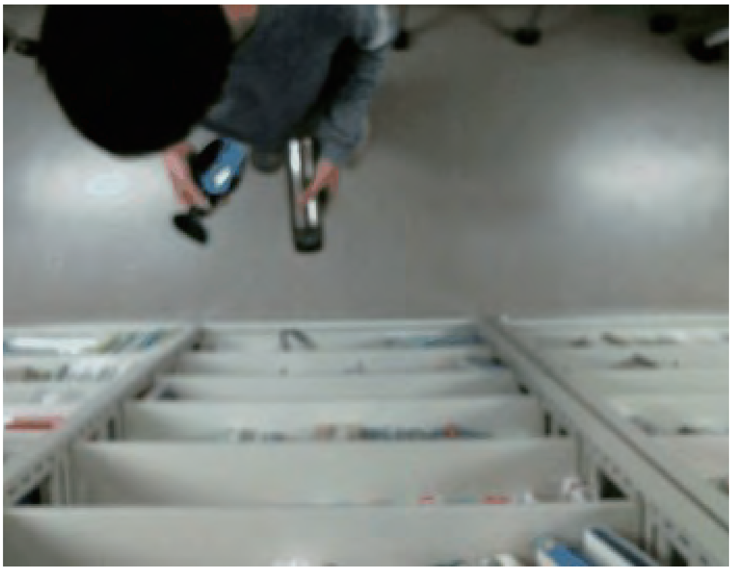


Figure 3.6: Example of our laboratory CAR dataset

### Experiment Settings

To evaluate the performance on adapting to environment changes, especially input video change and adding CAR models, we correspondingly took the experiments. Regarding the input video change, two datasets have videos with different resolutions, camera views, and camera distortion. Therefore, we apply both systems to our CAR dataset and then apply them to the MERL dataset. About adding CAR models, we simulate the environment change that leads to adding CAR models for new level's CA. The experiment follows the three steps below:

- **Step 1**: Need one CAR model for Lv.1 CA.
- **Step 2**: Need two CAR models for Lv.1 and Lv.2 CAs.
- **Step 3**: Need three CAR models for Lv.1, Lv.2, and Lv.4 CAs.

About the device information, we implemented both CAR systems on the same Windows 11 device with RAM of 16 GB. The CPU was an Intel i7-12700K (3.6 GHz). The GPU was an NVIDIA GeForce RTX 3060 Ti (8 GB). The program was written in Python 3.9. The ML framework was PyTorch 1.12. The used third-party libraries include numpy 1.22 and scipy 1.8.

Table 3.2: Adaptation cost comparison

| **Changes** | **Proposed CAR System** | **ML-based CAR System** |
|---|---|---|
| **Input Video Change** | Re-train level 1 model<br>(7 man-hours) | Re-train all CAR models<br>(21 man-hours) |
| **Add CAR Model(s)** | Step 1. Initialization: Add Level 1<br>Step 2. Add Level 2:<br>Sort [5]<br>(2 man-hours)<br>Step 3. Add Level 3:<br>Chapter 2 method<br>(114 man-hours) | Step 1. Initialization: Add Model 1<br>Step 2. Add Model 2:<br>YoloV4 [6]+Sort [5]<br>(106 + 2 man-hours)<br>Step 3. Add Model 3:<br>YoloV4 [6]+Sort [5]+LSTM [19]<br>(106 + 2 + 176 man-hours) |

### 3.4.3 Adaptation Cost Comparison

Table 3.2 compares the cost during the adaptation to environment changes. For the input video change, our proposed CAR system only re-trained a level 1 CAR model that is related to the input video. We spent about 7 man-hours. Existing ML-based CAR system re-trained all the CAR models, which took about 21 man-hours for the two CAR models.

Regarding adding CAR models, our proposed CAR system does not need to be modified since the outputs from the hierarchy cover the needed CAs. In the case of existing ML-based CAR system, it needs to add Model 1, 2 by annotating new target CAs and training Model 1 and Model 2. This time-consuming step took about 183 man-hours.

To sum up, the proposed CAR system can adapt to the input change easier with fewer modifications and less time. Also, the proposed system can be partially changed to adapt to environment changes related to input or any CA level. This indicates that the proposed CAR system is flexible.

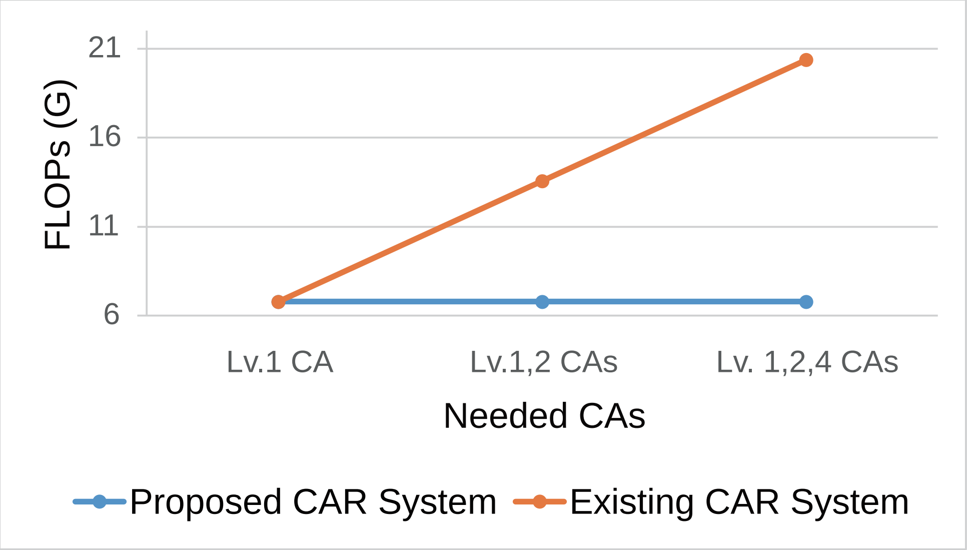


Figure 3.7: Change of FLOPs of both CAR systems during adding CAR models

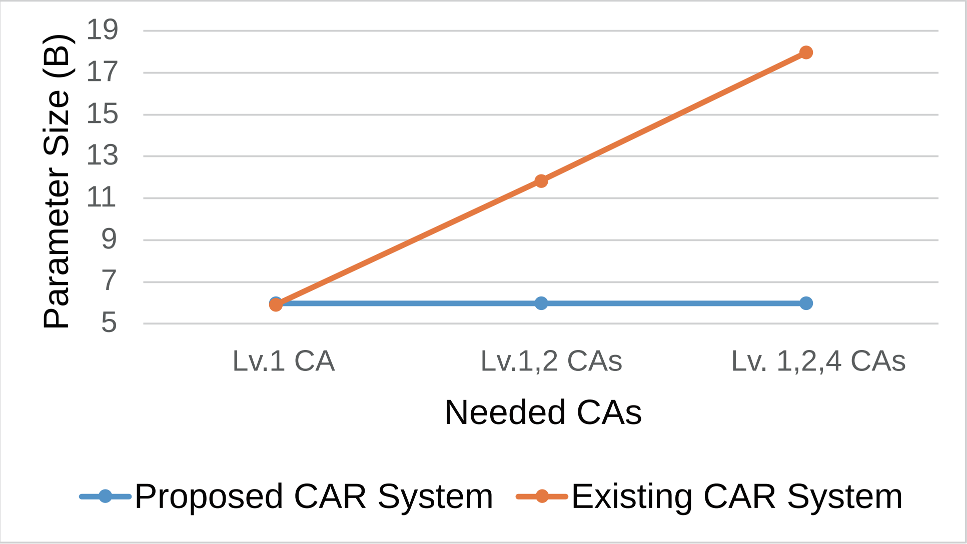


Figure 3.8: Change of parameter size of both CAR systems during adding CAR models

### 3.4.4 Performance Results

Figure 3.7 shows the change in Floating-point Operations Per Second (FLOPs) of both CAR systems during adding CAR models for new levels of CAs. Figure 3.8 shows the change of parameter size of both CAR systems during adding CAR models for new levels of CAs. FLOPs

Table 3.3: F1-score results

| CAR System | F1-score (MERL dataset) | F1-score (our CAR dataset) |
|---|---|---|
| Proposed CAR system | 83.9% | 92.0% |
| ML-based CAR system | 87.4% | 95.6% |

and parameter size reveal the amount of calculation, which reflects the performance of a CAR system. Results show that the performance of the existing CAR system is reducing with the increasing number of CAR models. On the other hand, our proposed CAR system has slight influence on performance when adding CAR models. Table 3.3 shows the f1-score of both CAR systems on different datasets. The Ml-based CAR models in existing CAR system are specialized for the target CBs, which leads to a good f1-score. However, the proposed CAR system is designed to be flexible enough to adapt to different environment changes, thus, it is not specialized for several target CBs, which leads to the decrease in accuracy. To improve the accuracy, the CAR models in low levels can be replaced by better methods to improve recognition accuracy. Another way for accuracy improvement is that more features can be extracted in low levels to provide more information to high levels, such as the orientation information. Moreover, in some cases that a customer moves in a so fast speed that leads to detection lost, one of the solutions is raising the frame rate of in-store cameras.

## 3.5 Structure Evaluation

### 3.5.1 Metric and Calculation

Except for the experiment that runs two systems, we design a score to evaluate a system's structure. The score is calculated from coupling, cohesion, and complexity [43], because these metrics reveal whether a system is flexible. To calculate these metrics, first we need to define the modules to be measured. In the proposed CAR system, we consider each level as a module, while in the case of existing CAR systems, each model is regarded as a module.

Based on the proposed formula in [43], we define the *coupling* as follows:

$$Coupling = 1 - \frac{1}{\max(1, w + r)}, \tag{3.1}$$

where $w$ is the count of calling other module's functions, $r$ is the count of functions called by other modules. For the existing system, Figure 3.4 shows that Model 2 has similar functions of object detection and tracking as Model 1. Thus, $w = 1, r = 0$ for Model 2 and $w = 0, r = 1$ for Model 1. For the hierarchy, no levels share similar functions. Therefore, $w = 0, r = 0$ for each level. After calculating the coupling of each module, the coupling of the whole system is defined as the average of all modules' coupling values.

In the case of *cohesion*, both systems are estimated to have high cohesion in their modules, which means no difference in this metric. Therefore, the cohesion calculation is omitted in the evaluation.

Finally, for the *complexity* calculation, we use a modified version of [43], where some irrelevant values were not included, as described below:

$$Complexity = IU + OU + MF, \tag{3.2}$$

where $IU$ is the count of required inputs from the user, $OU$ is the count of outputs from the system to the user, and $MF$ is the count of model files in the system. In our case, $IU$ and $OU$ indicate the count of data type instead of the amount of data. For instance, $IU = 1$ for the existing system because the input is an image. $IU = 2$ for the hierarchy because an extra input about the behavior definition is required. Finally, $MF$ refers to the number of models in both systems.

To normalize the complexity value, the formula is modified as follows:

$$Complexity' = \frac{(IU - 1) + (OU - 1) + (MF - 1)}{IU + OU + MF}, \tag{3.3}$$

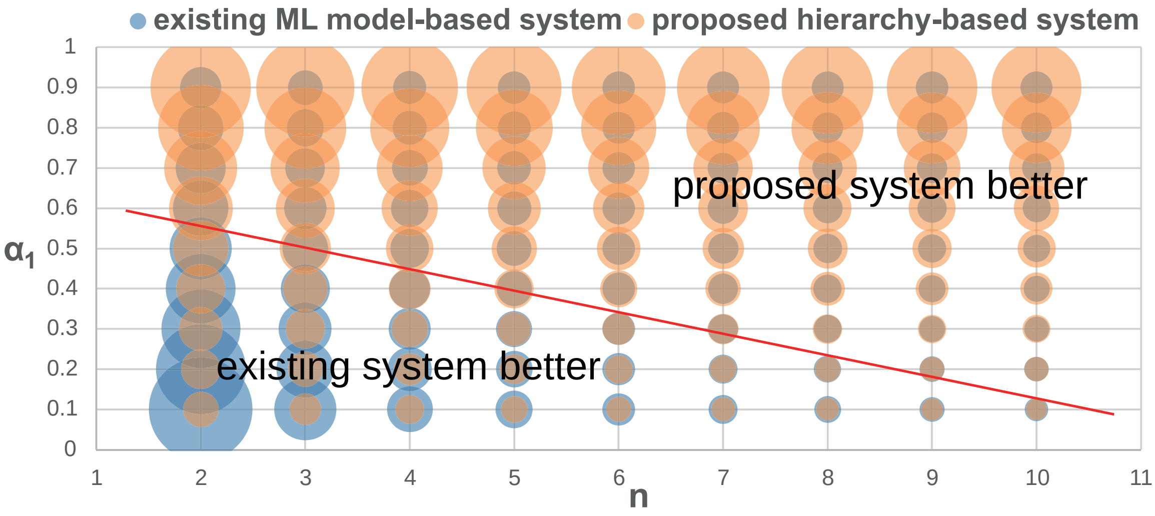


Figure 3.9: Structure evaluation results

where $Complexity'$ reaches the min value when $IU = 1$, $OU = 1$, $MF = 1$, and $Complexity'$ increases to a value no more than 1 when $IU$, $OU$, and $MF$ increase. The complexity of the whole system is defined as the average of all modules' complexity values.

Since a better system has lower coupling and complexity, we formulated the structure score (S) as

$$S = \frac{1}{\alpha_1 * Coupling + (1 - \alpha_1) * Complexity'}, \tag{3.4}$$

where $\alpha_1 \in [0, 1]$, $\alpha_1$ is the weight of $Coupling$ when designing a system, and a larger $\alpha_1$ means more emphasis on improving $Coupling$.

About the structure score, we set two parameters $(\alpha_1, n)$ to influence the score and observe its changes, where $n$ is the number of implemented models in each system. For the existing system, Figure 3.4 shows a simple system with only two models. However, usually, more models are required in a common CAR system. Therefore, $Coupling$ and $MF$ may change with the increasing number of models.

### 3.5.2 Structure Score Results

Figure 3.9 shows the change of the structure score when modifying $\alpha_1$ and $n$. The x-axis is $n$ and the y-axis is $\alpha_1$. The circle radius refers to the structure score. A larger circle represents a better structure score. The chart shows that our proposed system have a better score when $\alpha_1$ and $n$ increase. And, in most cases, the proposed system has a better score. Since a better structure score implies better flexibility, Figure 3.9 indicates that our proposed hierarchy-based system is better in most cases. To sum up, the structure score results show that the proposed hierarchy-based system has better flexibility to adapt to the environment changes.

## 3.6 Summary

In this chapter, we study a CAR system's adaptation to environment change. The objective is to realize a CAR system that is flexible to adapt to the environment changes that cause input change or adding CAR models. We proposed a hierarchy-based CAR system. It runs partial CAR models level by level to provide the CAs required by various retail purposes. The structure of the hierarchy avoids the overlapped processes among CAR models and the high coupling between inputs and CAR models. Therefore, the proposed CAR system achieves good flexibility when adapting to environment changes, which solves (P2) Hard to Adapt to Environment Change.

Despite of the achievement of better flexibility, recognition accuracy needs to be improved by applying better CAR methods to related levels of the hierarchy. Also, we believe that there are higher levels, such as intention, that could predict one's intention. A higher level could be able to support more retail purposes, like shoplifting prevention.

# 4 Conclusions

## 4.1 Summary

- **Objective**

  The objective of this study is to realize a CAR system that is flexible enough to adapt to changes for different retail purposes, especially (C1) Target Change and (C2) Environment Change. Regarding (C1), we particularly focus on the flexibility to adapt to the target CB change. Concerning (C2), we aim at the flexibility for input change and additional CAR models caused by environment change.

- **Proposal**

  In Chapter 2, we proposed a method to recognize CB from the video captured by in-store cameras. For each frame of the video, the proposed method recognizes CB using the combination of primitives, each of which is an entity's motion or a relationship between two entities. Only by changing the primitive combination is it possible to adapt to changes in target CBs with fewer modifications and in less time. In Chapter 3, we proposed a hierarchy-based CAR system to run partial CAR models along the four CA levels (Spatial, Temporal, Primitive, Behavior). When the environment change requires changing the input video, different inputs of different partial CAR models allow the system to be partially modified to adapt to input changes, which is flexible. Also, when the environment change needs to add more CAR models, non-overlapped processes make adding the models has slight influence on the performance of the CAR system.

- **Achievements**

  In Chapter 2, we applied our proposed primitive-based CBR method to different target CBs from two datasets. The results of the adaptation comparison show that our proposed method achieves flexibility when adapting to target CB changes, resolving (P1) Hard to Adapt to Target Change. In Chapter 3, we implemented our proposed hierarchy-based CAR system and an existing ML model-based CAR system. We changed different input videos from two datasets. Also, we added CAR models during the running of two CAR systems on two datasets. The results show that when adapting to input changes and adding CAR models caused by environment changes, our proposed CAR system achieves flexibility, which solves (P2) Hard to Adapt to Environment Change. As a result, by applying the primitive-based CBR method to the hierarchy-based CAR system, this study achieves a CAR system that is flexible for both (C1) and (C2).

## 4.2 Contributions

This study aims at a flexible CAR system in smart retail. With the aforementioned achievements, this study is expected to contribute to smart retail in the following ways, as shown in Figure 4.1.

- **Handle Target Change**:

  Since our proposed primitive-based CBR method is flexible enough to handle target changes easily, it is possible to recognize a wide range of CBs. Because CB is a kind of CA, the achievement of providing more CBs can be regarded as being able to (A) provide more CAs, which is shown in Figure 4.1.

- **Handle Environment Change (Changed Input)**

  The proposed hierarchy-based CAR system can flexibly adapt to the changed inputs. Therefore, the system is able to receive various sensor data. As a result, the system

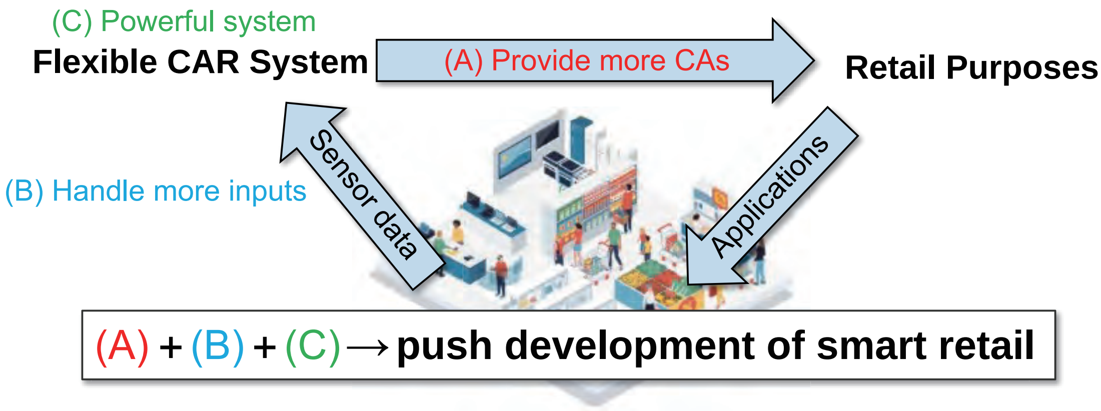


Figure 4.1: This study's contribution to smart retail.

can (B) handle more inputs.

- **Handle Environment Change (Increased CAR Models)**

  Also, when the environment change requires adding CAR models, our flexible CAR system can be easily expanded because the increased number of CAR models has slight influence on the system's performance. Consequently, for smart retail, it could be (C) powerful system.

Finally, for the left corner of the smart retail triangle in Figure 4.1, (A), (B), and (C) contribute a lot to the development of this corner. Later, it will drive the development of the other parts of smart retail. Therefore, the contribution of this study is to be able to greatly push the development of smart retail.

## 4.3 Future Directions

Although our proposed solutions solve existing problems, we still need more research to optimize current findings in the face of possible future problems or limitations of our proposed methods.

- **Primitive-Based CBR Method**

  Current primitives cannot define CBs that contain orientation information, such as face to, turn back to. However, the orientation information is essential to precisely defining some CBs. Thus, we will need to add more primitives for describing the orientation in the future.

  Besides, the current pattern matching algorithm only supports CB recognition. To support retail purposes such as shoplifting prevention, traffic prediction, CB prediction instead of recognition is better. Therefore, the CB prediction could also be one of the future directions.

- **Hierarchy-Based CAR System**

  Because CB prediction is required for retail purposes such as shoplifting prevention, traffic prediction, one future direction for improving our proposed CAR system could be to expand higher CA level intentions.

  Furthermore, the current CAR system is proposed to mainly receive video as input. Nevertheless, smart retail also has many other sensors installed. To achieve better flexibility, future work could involve improving the hierarchy to be compatible with other sensors.

# Acknowledgement

Throughout the whole time I spent writing this dissertation, I received a lot of help and support.

I would like to begin by acknowledging my supervisor, Prof. Takuo Suganuma, whose assistance in determining the objectives and methodology of this study was vital. In addition, I also want to extend my gratitude to Prof. Shinichiro Omachi and Prof. Xiao Zhou, whose insightful suggestions inspired me to improve my thinking and raised the quality of my work.

I would also want to thank my advisor, Assoc. Prof. Toru Abe, for his invaluable guidance throughout my graduate studies. You support me with the tools that I needed to choose the correct path and finish my dissertation successfully.

Furthermore, I could not have completed this dissertation without the support of the WISE Program for AI Electronics, which provided crucial funding for this study. Besides, the various courses and internship opportunities offered by the program have substantially enhanced my practical skills and abilities, allowing me to take a better view of this study.

Additionally, I would like to sincerely appreciate Prof. Kazunori Kotani for his constructive guidance on my study during my internship at the Japan Advanced Institute of Science and Technology (JAIST).

Moreover, I would particularly like to thank the lab's staff, Research Assoc. Luis Guillen and Midori Horino, as well as other lab members, for their excellent collaboration and patient support.

Finally, I would like to express my gratitude to my family and friends for their helpful advice and understanding ears. You are always there for me. Thank you for providing me with

encouraging conversations outside of my studies.

# Publications

## Journal Papers

**(1)** Jiahao Wen, Toru Abe, and Takuo Suganuma, “A customer behavior recognition method for flexibly adapting to target changes in retail stores,” *Sensors*, Vol.22, No.18, 6740, 2022.

**(2)** Jiahao Wen, Luis Guillen, Toru Abe, and Takuo Suganuma, “A hierarchy-based system for recognizing customer activity in retail environments,” *Sensors*, Vol.21, No.14, 4712, 2021.

## Conference Papers

**(3)** Jiahao Wen, Toru Abe, and Takuo Suganuma, “Customer behavior recognition adaptable for changing targets in retail environments, ” *18th IEEE International Conference on Advanced Video and Signal-Based Surveillance Hybrid (AVSS)*, Madrid, Spain, pp. 1-8, November-December, 2022.

**(4)** Jiahao Wen, Luis Guillen, Muhammad Alfian Amrizal, Toru Abe, and Takuo Suganuma, “An event-based hierarchical method for customer activity recognition in retail stores, ” *15th Advances in Visual Computing, International Symposium on Visual Computing (ISVC)*, San Diego, USA, pp.263–275, October, 2020.

## Others

**(5)** Jiahao Wen, Toru Abe, and Takuo Suganuma, “A malleable customer behavior recognition for different recognition targets in retail environments, ” *18th International Workshop on Emerging ICT (IWEICT)*, Online, October, 2021.

**(6)** Jiahao Wen, Luis Guillen, Muhammad Alfian Amrizal, Toru Abe, and Takuo Suganuma, “An event-based hierarchy to recognize customer activities in retail stores, ” *17th International Workshop on Emerging ICT (IWEICT)*, Online, October, 2020.

**(7)** Jiahao Wen, Muhammad Alfian Amrizal, Toru Abe, and Takuo Suganuma, “A flexible recognition method of customer activities in retail stores, ” 第23回 画像の認識・理解シンポジウム（MIRU2020）, Sendai, Japan, August, 2020. (Poster session)

# Reference

[1] Abdallah, T. B., Elleuch, I., and Guermazi, R., “Student behavior recognition in classroom using deep transfer learning with VGG-16,” *Procedia Computer Science*, Vol. 192, pp. 951-960, 2021.

[2] Agarap, A. F., “Deep learning using rectified linear units (relu),” *arXiv preprint arXiv:1803.08375*, 2018.

[3] Ahmad, M., Ahmed, I., Ullah, K., Khattak, A., and Adnan, A., “Person detection from overhead view: a survey,” *International Journal of Advanced Computer Science and Applications*, Vol. 10, No. 4, 2019.

[4] Akiba, T., Sano, S., Yanase, T., Ohta, T., and Koyama, M., “Optuna: a next-generation hyperparameter optimization framework,” *25th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 2623-2631, 2019.

[5] Bewley, A., Ge, Z., Ott, L., Ramos, F., and Upcroft, B., “Simple online and realtime tracking,” *2016 IEEE international conference on image processing (ICIP)*, pp. 3464-3468, 2016.

[6] Bochkovskiy, A., Wang, C. Y., and Liao, H. Y. M., “Yolov4: optimal speed and accuracy of object detection,” *arXiv preprint arXiv:2004.10934*, 2020.

[7] Boyer, R.S., and Moore, J.S., “A fast string searching algorithm,” *Communications of the ACM*, Vol. 20, No. 10, pp. 762-772, 1977.

[8] Candamo, J., Shreve, M., Goldgof, D. B., Sapper, D. B., and Kasturi, R., “Understanding transit scenes: a survey on human behavior-recognition algorithms,” *IEEE transactions on intelligent transportation systems*, Vol. 11, No. 1, pp. 206-224, 2009.

[9] Chawathe, S. S., “Beacon placement for indoor localization using bluetooth,” *11th International IEEE Conference on Intelligent Transportation Systems*, pp. 980-985, 2008.

[10] Chen, Y.Y., Zheng, Z.W., Chen, S.N., Sun, L., and Chen, D., “Mining customer preference in physical stores from interaction behavior, ” *IEEE Access*, Vol. 5, pp. 17436-17449, 2017.

[11] Cheng, B., Xiao, B., Wang, J.D., Shi, H.H., Huang, T.S., and Zhang, L., “Bottom-up higher-resolution networks for multi-person pose estimation,” *arXiv preprint arXiv:1908.10357*, Vol.7, 2019.

[12] Christodoulou, P., Christodoulou, K., and Andreou, A.S., “A real-time targeted recommender system for supermarkets,” *SciTePress*, 2017.

[13] Data Bridge Market Research, “Global smart retail market—industry trends and forecast to 2029,” (accessed on 18 June 2022), Available online: https://www.databridgemarketresearch.com/reports/global-smart-retail-market.

[14] de S. Silva D.V., de S. Silva R., and Durão F.A., “Recommending stores for shopping mall customers with recstore,” *Journal of Information and Data Management*, Vol. 9, No. 3, pp. 197-197, 2018.

[15] Fang, B., Liao, S.Y., Xu, K.Q., Cheng, H., Zhu, C., and Chen, H.P., “A novel mobile recommender system for indoor shopping,” *Expert Systems with Applications*, Vol. 39, No. 15, pp. 11992-12000, 2012.

[16] Frontoni, E., Raspa, P., Mancini, A., Zingaretti, P., and Placidi, V., “Customers’ activity recognition in intelligent retail environments,” *Image Analysis and Processing*, pp. 509-516, 2013.

[17] Generosi, A., Ceccacci, S., and Mengoni, M., “A deep learning-based system to track and analyze customer behavior in retail store,” *2018 IEEE 8th International Conference on Consumer Electronics-Berlin (ICCE-Berlin)*, pp. 1-6, 2018.

[18] Hernandez, D.A.M., Nalbach, O., and Werth, D., “How computer vision provides physical retail with a better view on customers,” *IEEE 21st Conf. Business Informatics (CBI)*, Vol. 1, pp. 462–471, 2019.

[19] Hochreiter, S., and Schmidhuber, J., “Long short-term memory,” *Neural computation*, Vol. 9, No. 8, pp. 1735-1780, 1997.

[20] IBM, “How industry 4.0 technologies are changing manufacturing,” (accessed on 19 Dec 2022), Available online: https://www.ibm.com/topics/industry-4-0.

[21] Kong, Y., and Fu, Y., “Human action recognition and prediction: a survey,” *International Journal of Computer Vision*, Vol. 130, No. 5, pp. 1366-1401, 2022.

[22] Lacic, E., Kowald, D., Traub, M., Luzhnica, G., Simon, J., and Lex, E., “Tackling cold-start users in recommender systems with indoor positioning systems,” *RecSys Posters*, 2015.

[23] Lee, K., Choo, C.Y., See, H. Q., Tan, Z. J., and Lee, Y., “Human detection using histogram of oriented gradients and human body ratio estimation,” *3rd International Conference on Computer Science and Information Technology*, Vol. 4, pp. 18-22, 2010.

[24] Leykin, A., and Tuceryan, M., “Detecting shopper groups in video sequences,” *IEEE Conference on Advanced Video and Signal Based Surveillance*, pp. 417-422, 2007.

[25] Liciotti, D., Contigiani, M., Frontoni, E., Mancini, A., Zingaretti, P., and Placidi, V., "Shopper analytics: a customer activity recognition system using a distributed RGB-D camera network," *International Workshop on Video Analytics for Audience Measurement in Retail and Digital*, pp. 146-157, 2014.

[26] Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C.L., "Microsoft coco: common objects in context," *European conference on computer vision*, pp. 740-755, 2014.

[27] Liu, H., Yale, H., and Tanja, S., "Motion units: generalized sequence modeling of human activities for sensor-based activity recognition," *29th European Signal Processing Conference (EUSIPCO)* ,pp.1506-1510, 2021.

[28] Liu, J., Liu, Y., Zhang, G., Zhu, P., and Chen, Y. Q., "Detecting and tracking people in real time with RGB-D camera," *Pattern Recognition Letters*, Vol. 53, pp. 16-23, 2015.

[29] Liu, J.W., Gu, Y.L., and Kamjjo, S.S., "Customer behavior recognition in retail store from surveillance camera," *IEEE International Symposium Multimedia*, pp. 154-159. 2015.

[30] Mansour, R.F., Escorcia-Gutierrez, J., Gamarra, M., Villanueva, J.A., and Leal, N., "Intelligent video anomaly detection and classification using faster RCNN with deep reinforcement learning model," *Image and Vision Computing*, Vol. 112, pp. 104229, 2021.

[31] Merad, D., Drap, P., Lufimpu-Luviya, Y., Iguernaissi, R., and Fertil, B., "Purchase behavior analysis through gaze and gesture observation," *Pattern Recognition Letter 2016*, Vol. 81, pp. 21–29, 2016.

[32] Minto, B., "MECE: I invented it, so I get to say how to pronounce it," (accessed on 9 Jan 2023), Available online: https://www.mckinsey.com/alumni/news-and-events/global-news/alumni-news/barbara-minto-mece-i-invented-it-so-i-get-to-say-how-to-pronounce-it.

[33] Paolanti, M., Liciotti, D., Pietrini, R., Mancini, A., and Frontoni, E., "Modelling and forecasting customer navigation in intelligent retail environments," *Journal of Intelligent & Robotic Systems*, Vol. 91, No. 2, pp. 165-180, 2018.

[34] Paolanti, M., Pietrini, R., Mancini, A., Frontoni, E., and Zingaretti, P., "Deep understanding of shopper behaviours and interactions using RGB-D vision," *Machine Vision and Applications*, Vol. 31, No. 66, pp. 1-21, 2020

[35] Popa M.C., Gritti T., Rothkrantz L. J. M., Shan C.F., and Wiggers P., "Detecting customers' buying events on a real-life database," *International Conference Computer Analysis of Images and Patterns (CAIP), Vol. 6854, pp. 17-25, 2011*

[36] Popa, M.C., Rothkrantz, L.J.M., Wiggers, P., and Shan, C., "Shopping behavior recognition using a language modeling analogy," *Pattern Recognition Letters*, Vol. 34, No. 15, pp. 1879-1889, 2013.

[37] Popescu, G., "Human behavior, from psychology to a transdisciplinary insight," *Procedia-Social and Behavioral Sciences*, Vol. 128, pp. 442-446, 2014.

[38] Rai, N., Chen, H., Ji, J., Desai, R., Kozuka, K., Ishizaka, S., Adeli, E., and Niebles, J.C., "Home action genome: cooperative compositional action understanding," *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11184-11193, 2021.

[39] Singh, B., Marks, T.K., Jones, M., Tuzel, O., and Shao, M, "A multi-stream bi-directional recurrent neural network for fine-grained action detection," *2016 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 1961–1970. 2016.

[40] So, W.T., and Yada, K., "A framework of recommendation system based on in-store behavior, " *4th Multidisciplinary International Social Networks Conference (MISNC), Vol. 33, pp. 1-4, 2017.*

[41] Sturari, M., Liciotti, D., Pierdicca, R., Frontoni, E., Mancini, A., Contigiani, M., and Zingaretti, P, "Robust and affordable retail customer profiling by vision and radio beacon sensor fusion," *Pattern Recognition Letters*, Vol. 81, pp. 30-40, 2016.

[42] Szwacka-Mokrzycka, J., "Trends in consumer behaviour changes: overview of concepts," *Acta Scientiarum Polonorum. Oeconomia*, Vol. 14, No. 3, pp. 149-156, 2015.

[43] Vishnyakov, A., and Orlov, S., "Software architecture and detailed design evaluation," *Procedia Computer Science*, Vol. 43, pp. 41-52, 2015.

[44] Waqas, M., Nasir, M., Samdani, A.H., Naz, H., and Tanveer, M., "Customer activity recognition system using image processing," *International Journal of Computer Science & Network Security*, Vol. 21, No. 9, pp. 63-66, 2021.

[45] Wen, J., Guillen, L., Abe, T., and Suganuma, T., "A hierarchy-based system for recognizing customer activity in retail environments," *Sensors*, Vol.21, No.14, 4712, 2021.

[46] Wikipedia, "Behavior," (accessed on 22 Dec 2022), Available online: https://en.wikipedia.org/wiki/Behavior#cite_note-Szwacka-11.

[47] Wikipedia, "Fourth industrial revolution," (accessed on 19 Dec 2022), Available online: https://en.wikipedia.org/wiki/Fourth_Industrial_Revolution.

[48] Yale H., Liu H., Steffen L., and Tanja S., "Interpretable high-level features for human activity recognition," *15th International Joint Conference on Biomedical Engineering Systems and Technologies - BIOSIGNALS*, pp. 40-49, 2022.

[49] Yamamoto, J., Inoue, K., and Yoshioka, M., "Investigation of customer behavior analysis based on top-view depth camera," *IEEE Winter Applications of Computer Vision Workshops (WACVW)*, pp. 67-74, 2017.

[50] Zhang, S., and Wang, X., "Human detection and object tracking based on Histograms of Oriented Gradients," *9th international conference on natural computation*, pp. 1349-1353, 2013.

[51] Zhao, L., Yao, J., Du, H., Zhao, J., Zhang, R., "A unified object detection framework for intelligent retail container commodities," *IEEE International Conference on Image Processing (ICIP)*, pp. 3891-3895, 2019.

[52] Zheng, Z., Chen, Y., Chen, S., Sun, L., and Chen, D., "Location-aware POI recommendation for indoor space by exploiting WiFi logs," *Mobile Information Systems*, 2017.