Cross-view Semantic Alignment for Livestreaming Product Recognition

Wenjie Yang*, Yiyi Chen*, Yan Li, Yanhua Cheng, Xudong Liu, Quan Chen†, Han Li
Kuaishou Technology

wenjie.yang@nlpr.ia.ac.cn, {chenyiyi,liyan26,chengyanhua,liuxudong,chenquan06,lihan08}@kuaishou.com

Abstract

Live commerce is the act of selling products online through live streaming. The customer’s diverse demands for online products introduce more challenges to Livestreaming Product Recognition. Previous works have primarily focused on fashion clothing data or utilize single-modal input, which does not reflect the real-world scenario where multimodal data from various categories are present. In this paper, we present LPR4M, a large-scale multimodal dataset that covers 34 categories, comprises 3 modalities (image, video, and text), and is 50× larger than the largest publicly available dataset. LPR4M contains diverse videos and noise modality pairs while exhibiting a long-tailed distribution, resembling real-world problems. Moreover, a cRoSS-vieW semanTic alignmEnt (RICE) model is proposed to learn discriminative instance features from the image and video views of the products. This is achieved through instance-level contrastive learning and cross-view patch-level feature propagation. A novel Patch Feature Reconstruction loss is proposed to penalize the semantic misalignment between cross-view patches. Extensive experiments demonstrate the effectiveness of RICE and provide insights into the importance of dataset diversity and expressivity. The dataset and code are available at https://github.com/adxcreative/RICE.

1. Introduction

Livestreaming Product Recognition (LPR) [3, 9, 11] is one of the significant machine learning application in the e-commerce industry. Its goal is to recognize products a salesperson presents in a live commerce clip through content-based video-to-image retrieval. The real-time and accurate recognition of livestreaming products can facilitate the online product recommendation, and thereby improve the purchasing efficiency of consumers.

The task of LPR involves two fundamental processes: multimodal-based intended product identification and shop product retrieval. This task poses significant challenges in real-world scenarios, including (1) the need to distinguish intended products from the cluttered background products in a livestreaming frame, exemplified in Fig. 1(b), (2) the requirement for models to capture sufficient fine-grained features to match the ground-truth (GT) image accurately in the shop, where there are many images with subtle visual nuances, (3) the heterogeneous video-to-image and cross-domain livestream-to-shop problem, and (4) the appearance changes of products in the livestreaming domain due to articulated deformations, occlusions, diverse background clutters, and significant illumination variations, making it a highly intricate task to match the clip to the GT image in the shop. Various datasets have emerged in the computer vision community to study this task, including AsymNet [3], WAB†, and MovingFashion [9]. However, AsymNet and MovingFashion lack crucial text modal, which provides essential auxiliary information for identifying intended products. Furthermore, the data scale of WAB is relatively small, with only 70K pairs, and only provides fashion clothing data, diverging from the real-world scenario.

In order to narrow the gap between existing datasets...
and the real-world scenario and advance research in this challenging task, we present LPR4M, a large-scale multi-modal live commerce dataset that includes extensive categories, diverse data modalities of clip, image, and text, as well as heterogeneous and cross-domain correspondences of \(\langle\text{clip, image}\rangle\) pairs. This dataset offers several significant advantages. (1) Large-Scale: LPR4M contains over 4M pairs, significantly exceeding its precedents. (2) Expressivity: LPR4M draws data pairs from 34 commonly used live commerce categories rather than relying solely on clothing data. Additionally, LPR4M offers auxiliary clip ASR text and image title modalities, which are critical for intended product identification and product feature representation. (3) Diversity: LPR4M promotes clip diversity while preserving the real-world data distribution, with a focus on three components: product scale, visible duration, and the number of products in the clip, as depicted in Fig. 3. To the best of our knowledge, LPR4M is currently the largest dataset created explicitly for real-world multi-modal LPR scenarios.

Our work based on LPR4M tackles a realistic problem: how to achieve fine-grained LPR using large-scale multi-modal pairwise data? Given image and clip views, we first utilize Instance-level Contrastive Learning (ICL) to align global features. However, since instance features of these two views are extracted independently from the visual encoder, it can be challenging to differentiate between products with subtle visual differences without cross-view interactions. Consequently, we propose a patch-level semantic alignment approach to enable cross-view patch information propagation. We suggest measuring similarity via a cross-attention based Pairwise Matching Decoder (PMD), which treats image patches as Query and video patches as both Key and Value. In addition, we propose a novel Patch Feature Reconstruction (PFR) loss to penalize the semantic misalignment between patches of video and image. (4) The benchmark dataset and evaluation protocols are carefully defined for LPR. Extensive experiments demonstrate the effectiveness of LPR4M and RICE.

2. Related Works

LPR datasets. As shown in the table of Fig. 2, we compare LPR4M with others in terms of modality, content, and scale. In particular, AsymNet [3], WAB, and MovingFashion [9] only provide fashion clothing data, and the text modality is absent in AsymNet and MovingFashion. Therefore, We collect LPR4M, which covers 34 widely used categories and provides visual and text modalities. LPR4M is 50× larger than WAB.

Video Object Detection (VOD). The main focus of recent VOD methods [38, 37, 4, 2, 26, 35] is exploiting temporal information to tackle the video variations, e.g., oc-
clusion, motion blur and out of focus. The temporal relationship mining insights in VOD inspire this paper’s design of the intended product detection module. However, unlike VOD, where most videos only contain a single object, the videos in LPR contain many cluttered background products. This situation suggests that in the absence of prompt text information, it is more challenging to identify the intended product by only relying on the visual inputs. Therefore, we explore the fusion of text and visual modalities and verify its effectiveness in this paper.

Fine-Grained Vision Recognition (FGVR). Similar to FGVR [6, 34, 5, 16, 7, 36, 30, 29], the LPR aims to learn discriminative instance feature to distinguish the subclasses with large intra-class and small inter-class variations. However, unlike traditional FGVR, each live commerce category contains an enormous amount of subclasses in LPR. Moreover, the number of subclasses will increase or decrease dynamically as a large number of products are newly added or taken off the shop every day. It makes it more challenging to handle the out-of-distribution subclasses.

Video-to-Shop Retrieval. Although fashion retrieval has made great progress [13, 19, 8, 15], there are few studies focus on retrieving products that are presented in e-commerce videos, referred to as video-to-shop. AsymNet is a one-stage method without detection. It employs LSTM to exploit temporal continuity in the video, then perform pair-wise matching by feeding the image and video feature into a similarity network. DPRNet [33] and SEAM Match-RCNN [9] adopt a two-stage pipelines. DPRNet first detects the products in the video and then performs image-to-image retrieval. SEAM Match-RCNN performs self-attention among the detected product boxes in a video to produce a video feature and uses the inner product between the video and image feature as a similarity. In this paper, we propose RICE integrates the one-stage and two-stage methods into a framework and study their advantages.

3. Dataset and Benchmark

In this section, we present the construction, characteristics, and benchmark of LPR4M.

Overview. Compared with other existing LPR datasets, LPR4M has several appealing properties, which are summarized in the following. (1) Large-Scale. As illustrated in the table of Fig. 2, LPR4M is the largest LPR dataset to date. It contains 4M exactly matched ⟨clip, image⟩ pairs of 4M live clips, and 332k shop images. Each image has 14.5 clips with different product variations, e.g., viewpoint, scale, and occlusion. The example of image-to-clips and the number of clips per image are shown in Fig. 4 (d) and (b), respectively. Specifically, most of the images (80%) have ten matched clips and the number of clips per image range from 10 to 150. (2) Expressivity. The expressivity of LPR4M is mainly reflected in two aspects. Firstly, unlike other LPR datasets that only contain fashion clothing data, our data is more affluent, coming from 34 categories covering most of the daily necessities. This makes it closer to the real scenario. Secondly, the data of LPR4M is multimodal. We provide live clip ASR texts and shop image titles as auxiliary information to facilitate the intended product identification and form a full-scale characteristic of each product. (3) Diversity. Firstly, we collect clips according to the clip duration distribution of real livestreaming scenarios and obtain the clips with various durations, as shown in Fig. 4 (a). Secondly, the clips are further sampled by controlling the variation in terms of three properties, i.e., product scale, intended product visible duration, and the number of products in the clip. It makes LPR4M a challenging benchmark. As illustrated in Fig. 3, we pick two categories to represent a variation. For each category row, the clip shows three different levels of difficulty progressively.

3.1. Data Collection and Cleaning

The basic unit of the dataset is a ⟨clip, image⟩ pair. All the clips are cut from hours of sequential livestreaming data crawled from Kuaishou2. A livesteaming has a unique online shop, which lists all the products to be introduced in this livestreaming. Firstly, we removed near- and exact-duplicate images in the shops by comparing the global average pooled layer4 features after feeding them into ResNet [12]. Secondly, the human annotators cleaned the clips that contain the target product with short visible duration, small scales, and severe background clutters. Finally, given a clip, the matched product image is picked from the shop via the human annotator. In total, 4,033,696 clips and 398,796 images are kept to construct the training and test set of LPR4M.

Variations. The proportion of the number of clips in each variation is depicted in Fig. 4 (c). (1) Scale. According to the proportion (p) of the product box area to the entire frame area, the clips are classified into three subsets. The area is measured as the number of pixels in the product box. In LPR4M, there are more small products than large products. Specifically, approximately 54.5% of products are small (p≤0.2), 30.5% are medium (0.2<p≤0.4), and 15% are large (p>0.4). As shown in the first row of Fig. 3, the coat in the first clip is displayed in a zoom-in view, where the coat is large and overflows the frame. However, the physical size of the shoes in the third clip is relatively small. Because the considerable distance between the shoes and the camera, the scale is visually small. (2) Visible duration. Due to occlusions and changes in camera perspective, the target product is not always visible in the clip. Here, each clip is categorized by the proportion of visible duration to the entire clip duration, including 48.5% of long (0.7<p), 29.6% of medium (0.4<p≤0.7) and 21.9% of

2https://live.kuaishou.com
Figure 3. The \(<\text{clip, image}>\) pairs of LPR4M. As shown on the left of the orange box, we extract five evenly spaced frames from the clip, with the shop image on the right. We choose two categories to illustrate one of the clip product variations of \textit{scale}, \textit{visible duration}, and \textit{number of products}. Each row shows 3 data pairs for different degrees of difficulty of the corresponding variation, including (a) \textit{large}, \textit{medium} and \textit{small} product scale, (b) \textit{long}, \textit{medium} and \textit{short} visible duration, (c) \textit{abundant}, \textit{medium} and \textit{few} products in the clip.

Figure 4. LPR4M statistics of clips. (a) The distribution of clip duration. (b) The number of clips per image. Approximately 80\% of the images have ten paired clips. (c) The statistics of three clip variations. (d) The shop image, the paired clips, and the products in the clips suffer from different variations, e.g., scale, viewpoint, and occlusion.

short \((p \leq 0.4)\). For example, in the third clip of the fourth row in Fig. 3, the watch is occluded at the beginning and end of the clip, which significantly increases the difficulty of LPR. Note that the visible duration of the intended product is evaluated by the annotators. (3) \textbf{Background distractor.} In the livestreaming of beauty makeup, handbags, and jewelry, etc., there are abundant products displayed on the screen. For example, the first clip in the last row of Fig. 3 contains more than two dozen perfumes. However, there is only one intended product in a clip, and it is challenging to distinguish the intended product from the distracted background products. Therefore, we asked the annotators to assess the number \((n)\) of products in the clip (or background distractor) and accordingly classify the clips into three subsets, including 13.6\% of abundant \((n>7)\), 45.8\% of medium \((3<n \leq 7)\) and 40.6\% of few \((n \leq 3)\).

\textbf{Clip Description and Image Title.} In the case of a clip containing multiple products, it is ambiguous for the model to predict whether the clip and an image match based on visual information only. Therefore, we additionally provide clip descriptions and image titles to enrich the dataset. On the one hand, benefiting from promising results achieved by the Transformer and Convolution Neural Network based models in ASR [10, 32, 31], we adopt Conformer [10], a SOTA method on the widely used LibriSpeech benchmark [23], to extract text description from the clip voices.
On the other hand, the titles of product images are provided by the merchant and are available on the video website.

3.2. Livestreaming Product Recognition

As shown in Fig. 1, this task is to retrieve the GT images from the shop (gallery) for each livestreaming clip (query) and has been considered by several previous works [13, 11, 19, 8, 3, 9]. It emphasizes the retrieval performance and considers the impact of intended product identification. Specifically, a query clip is counted as missed if the intended product fails to be identified. Rank-\(k\) retrieval accuracy is used to measure retrieval performance, such that a successful retrieval is counted if the GT image has been retrieved in the rank-\(k\) results.

**Splitting training and test set.** In order to evaluate different methods, we split the training and test set and ensure the products in the training and test set are non-overlapping. The training and test sets contain 4,013,617/332,438 and 20,079/66,358 clips/images, respectively.

**Intended product box annotation.** To enable effective supervision of the detector training and evaluation of the detection accuracy, we annotate the intended product box for both the training and test set. For the training set, we sample 2% of the clips for intended box annotation and extract 10 frames at even intervals. For each test clip, we extract one frame every 3 seconds. The detection training/test set contains 1,120,410/501,656 frames with 1,115,629/669,374 intended product boxes, respectively.

4. Method

This section presents the technical details of RICE. As shown in Fig. 5, the RICE first performs instance-level contrastive learning to learn discriminative feature for the product (Sec. 4.1). Then, we introduce PMD that pursues fine-grained similarity measurement by conducting patch-level feature propagation (Sec. 4.2). The PMD is further guided by the novel PFR loss to promote patch-level semantic alignment (Sec. 4.2). Finally, we study the impact of product location by replacing the input patches with the product boxes produced by intended product detector (Sec. 4.3).

4.1. Instance-level Contrastive Learning (ICL)

Let \( \mathcal{V} \) be a set of livestreaming clips (or videos). Let \( \mathcal{I} \) be a set of shop images. The objective of RICE is to learn a function to measure the similarity between the clip \( V_i \in \mathcal{V} \) and the image \( I_i \in \mathcal{I} \). Formally, taking \( V_i \) and \( I_i \) as input, the image encoder first splits the image into non-overlapping patches, which are projected into 1D tokens via
a linear projection. Then the transformer layers are used to extract the patch features, denoted as \( \{ i_{cls}, i_1, ..., i_N \} \). Likewise, the video encoder processes each video frame \( V_j^i \) independently and outputs a sequence of video patch features \( \{ v_{cls}, v_1, ..., v_M \} \), where \( j \) is the index of frame number \( |V_j| \) and \( M = N \times |V_j| \). Given an image with a resolution of \( 224 \times 224 \) and patch size of \( 32 \times 32 \), we have \( N = 49 \). Note that the image and video encoder share parameters. Following ViT and CLIP, we extract the global representation from the [CLS] token. In order to pull the clip and image of the same product while pushing away that of different products in the feature space, we perform InfoNCE loss \([22]\) on the global representation, defined as:

\[
L_{nce} = - \mathbb{E}_{p(I, V)} \left[ \log \frac{\exp(g_\theta(I_i, V_i))}{\sum_{\tilde{V}_k \in \tilde{V}} \exp(g_\theta(I_i, \tilde{V}_k))} \right],
\]

where \( g_\theta(I_i, V_j) = g_\tau(i_{cls})^T g_\tau(v_{cls}) / \tau \) and \( \tilde{V} \) consists of a positive sample \( V_i \) and \( |\tilde{V}| - 1 \) negative samples. \( g_\tau \) and \( g_\tau \) are transformations that map the [CLS] embedding of image and clip, i.e., \( i_{cls} \) and \( v_{cls} \), to the normalized lower-dimensional features. \( \tau \) is a temperature parameter, and we use \( \tau = 0.01 \). The final contrastive loss between the image and clip is a symmetric version of \( L_{nce} \), given by:

\[
L_c = - \frac{1}{2} \mathbb{E}_{p(I, V)} \left[ \log \frac{\exp(g_\theta(I_i, V_i))}{\sum_{\tilde{V}_k \in \tilde{V}} \exp(g_\theta(I_i, \tilde{V}_k))} + \log \frac{\exp(g_\theta(I_i, V_i))}{\sum_{\tilde{I}_k \in \tilde{I}} \exp(g_\theta(\tilde{I}_k, V_i))} \right],
\]

where \( |\tilde{I}| = |\tilde{V}| \) is the batch size.

### 4.2. Patch-level Semantic Alignment

#### Pairwise Matching Decoder (PMD).

It is straightforward to exploit \( g_\theta(I_i, V_j) \) in Eq. (1) as a measure of the similarity. However, the features of the clip and image are extracted independently from the visual encoder, without information propagation between \( I_i \) and \( V_j \). To this end, we perform patch-wise feature attention via a transformer decoder layer, named pairwise matching decoder, which consists of a self-attention and a cross-attention layer in this paper. As illustrated in Fig. 5 (b), the self-attention layer takes the image patch features as the Query, Key and Value, and the cross-attention layer takes the image patch features as Query while takes the video patch features as Key and Value. The matching loss of the similarity calculator is defined as follows:

\[
L_m = - \frac{1}{2} \mathbb{E}_{p(I, V)} \left[ \log \frac{\exp(f_\beta(I_i, V_j))}{\sum_{\tilde{V}_k \in \tilde{V}} \exp(f_\beta(I_i, \tilde{V}_k))} + \log \frac{\exp(f_\beta(I_i, V_j))}{\sum_{\tilde{I}_k \in \tilde{I}} \exp(f_\beta(\tilde{I}_k, V_i))} \right],
\]

where \( f_\beta \) is the matching loss of the similarity calculator.

#### Patch Feature Reconstruction (PFR).

Furthermore, we pursue cross-view semantic alignment by searching similar patches in the clip to reconstruct the coupled image in the feature space. Here, we introduce how to perform patch feature reconstruction given a positive data point of two views of clip \( V_i \) and image \( I_i \). As shown in Fig. 5 (b), let

\[
X = \{ v_1, ..., v_M \} \in \mathbb{R}^{d \times M}
\]

be the patch features of the clip, where \( v_m \in \mathbb{R}^{d \times 1} \). Likewise, the patch features of the image are denoted as:

\[
Y = \{ i_1, ..., i_N \} \in \mathbb{R}^{d \times N}
\]

Then, the \( i_n \) can be represented by a linear combination of \( X \). The insight behind this is that the image can be reconstructed from the clip if the clip contains the product in the image. Therefore, we solve for the coefficients \( w_n \in \mathbb{R}^{M \times 1} \) of \( i_n \) with respect to \( X \). Finally, the reconstruction loss is defined as:

\[
L_r = \| Y - X W \|_F^2.
\]

Since the attention weights \( \alpha \) in the cross-attention layer indicate the correspondences between patches of the two views, it is intuitive to learn the reconstruction coefficients \( W \) from \( \alpha \). Specifically, the \( \alpha \in \mathbb{R}^{8 \times N \times M} \) is fed into two consecutive sets of convolution and ReLU layers to output the coefficients \( W \in \mathbb{R}^{N \times M} \).

The final objective function for the RICE model is the weighted summation of \( L_c, L_m \) and \( L_r \), given by:

\[
L = L_c + L_m + \alpha L_r,
\]

where \( \alpha \) is the trade-off weight and we use \( \alpha = 0.1 \) in the following experiments.

![Figure 6. Illustration of the Single-Frame Detector (SD) and Multi-Frame Detector (MD). The MD exploits a temporal transformer to fuse the results from the SD.](image-url)
Table 1. The R1 performances of the ICL and RICE model. Results of evaluation on different input types, i.e., patch and detected box, are shown in each row. The columns show the results on different test subsets split by video variations, i.e., product scale, visible duration, number of products. The best performance for each subset is in bold.

<table>
<thead>
<tr>
<th>Methods</th>
<th>overall</th>
<th>small</th>
<th>medium</th>
<th>large</th>
<th>short</th>
<th>medium</th>
<th>long</th>
<th>abundant</th>
<th>medium</th>
<th>few</th>
</tr>
</thead>
<tbody>
<tr>
<td>ICL\textsubscript{patch}</td>
<td>27.1</td>
<td>23.9</td>
<td>34.4</td>
<td>30.0</td>
<td>23.6</td>
<td>28.8</td>
<td>36.3</td>
<td>20.2</td>
<td>25.0</td>
<td>27.5</td>
</tr>
<tr>
<td>ICL\textsubscript{box}</td>
<td>30.0</td>
<td>30.1</td>
<td>35.8</td>
<td>31.1</td>
<td>26.6</td>
<td>30.1</td>
<td>37.1</td>
<td>17.2</td>
<td>29.2</td>
<td>30.6</td>
</tr>
<tr>
<td>RICE\textsubscript{patch}</td>
<td>31.2</td>
<td>28.9</td>
<td>37.0</td>
<td>32.7</td>
<td>28.1</td>
<td>32.9</td>
<td>39.6</td>
<td>21.0</td>
<td>34.6</td>
<td>31.5</td>
</tr>
<tr>
<td>RICE\textsubscript{box}</td>
<td>33.0</td>
<td>32.7</td>
<td>39.0</td>
<td>33.8</td>
<td>29.6</td>
<td>34.8</td>
<td>42.0</td>
<td>17.6</td>
<td>31.9</td>
<td>33.4</td>
</tr>
</tbody>
</table>

4.3. Intended Product Detection (IPD)

In order to highlight the intended products in video and suppress the background products, we propose replacing patch inputs with the detected intended product boxes for the videos. As shown in Fig. 6, we adopt DAB-DETR [18] and TransVOD Lite [35] as the SD and MD, respectively. The SD detects products frame-by-frame. Given the $T$ frames and one box label per frame, the SD is trained to predict an intended product box for each frame. However, detecting products with significant appearance changes using only a single frame can be challenging, as exemplified by the shoe with small scale in $V^T$ in Fig. 6. Therefore, MD leverages a temporal transformer to capture product interactions in the temporal context and predict a more accurate box for each frame. For more details about the IPD, please refer to the supplementary material.

5. Experiment

5.1. Dataset and Evaluation Metrics

Experiments are performed on the LPR4M testset, which contains 20,079 livestreaming clips as query set and 66,358 shop images as gallery set, as described in Sec. 3.2. We adopt rank-$k$ accuracy as the retrieval performance metrics.

5.2. Implementation Details

Model. The image and video encoder share parameters and are initialized with ViT-B/32 from CLIP [25], where the number of layers is 12 and the patch size is 32. Likewise, we initialize PDM with the similar parameters from CLIP.

Preprocessing. We extract 10 evenly spaced frames from each clip as the video input. The images and video frames are resized to $224 \times 224$. For data augmentation, we randomly mask video frames with a percentage ranging from 0 to 0.9 and a probability of 0.5. Optimization. We use Pytorch [24] to implement the RICE model. The Adam [14] optimizer is used with a batch size of 256. For the learning rate, we decay it using a cosine schedule [21] following CLIP. The initial learning rate is $1e-7$ for the image encoder and video encoder and $1e-4$ for the newly introduced modules. All the experiments are carried out on 8 NVIDIA Tesla V100 GPUs, which takes about 90 hours for 3 epochs.

Dataset | Methods | R1 | R5 | R10
---|---------|----|----|----
LPR4M  | NVAN [17] | 21.4 | 45.2 | 62.7
      | TimeSFormer [1] | 28.6 | 56.8 | 69.0
      | SwinB [20] | 29.1 | 60.1 | 73.9
      | RICE (Ours) | 33.0 | 65.5 | 77.3
MF     | NVAN [17] | 38.0 | 62.0 | 70.0
      | MGH [27] | 40.0 | 59.0 | 66.0
      | AsymNet [3] | 42.0 | 73.0 | 86.0
      | SEAM [9] | 49.0 | 80.0 | 89.0
      | RICE (Ours) | 76.1 | 89.7 | 92.6

5.4. Comparison with state-of-the-art methods

In this section, we compare our RICE with state-of-the-art (SOTA) methods on LPR4M and MovingFashion (MF), except AsymNet [3] and WAB because AsymNet is not public available and WAB is a competition dataset with only Chinese introduction. The results are shown in Table 2. 1) On LPR4M, the FashionNet, AsymNet and SEAM are LPR methods and the others are video understanding methods. As we can see, our RICE surpasses not only the LPR meth-
Table 3. Ablation study on the key components, i.e., ICL: instance-level contrastive learning, PMD: pairwise matching decoder, PFR: patch feature reconstruction, IPD: intended product detection, Txt: text modality. The rank-\(k\) accuracy is reported.

<table>
<thead>
<tr>
<th>#</th>
<th>ICL</th>
<th>PMD</th>
<th>PFR</th>
<th>IPD</th>
<th>Txt</th>
<th>R1</th>
<th>R5</th>
<th>R10</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>27.1</td>
<td>56.4</td>
<td>68.3</td>
</tr>
<tr>
<td>b</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>28.5</td>
<td>58.9</td>
<td>71.5</td>
</tr>
<tr>
<td>c</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>29.4</td>
<td>62.0</td>
<td>73.7</td>
</tr>
<tr>
<td>d</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>30.3</td>
<td>62.7</td>
<td>74.0</td>
</tr>
<tr>
<td>e</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>31.3</td>
<td>63.2</td>
<td>74.3</td>
</tr>
<tr>
<td>f</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>33.0</td>
<td>65.5</td>
<td>77.3</td>
</tr>
</tbody>
</table>

In this section, we investigate the impact of each component of our approach by conducting ablation experiments. The results are reported in Table 3. (c) Compared to the baseline ICL, the PMD obtains the R1 performance gains of 2.3% (29.4 to 27.1), which demonstrates the superiority of patch-level (local) over instance-level (global) similarity measurement. (d) The patch-level supervision provided by PFR facilitates semantic alignment and results in a considerable improvement of 0.9% for R1. (e) Our IPD replacing patch inputs with detected intended boxes significantly outperforms ICL by 1.0% R1 as it enables the model to focus on informative regions while suppressing distractions. In (b) and (f), the addition of text modality increases the R1 from 27.1% to 28.5% and 31.3% to 33.0%, respectively. It is because the text helps suppress the distracted background products. Here, the ChineseCLIP [28] is used to extract the embeddings of video ASR and image titles. The text similarity is computed as the dot product of normalized features. Then we combine the text and visual similarities to obtain the final \(\langle clip, image \rangle\) similarity via addition.

5.6. Per-category performance

As shown in Fig. 7, we compare the rank-1 accuracy of FashionNet [19], SwinB [20] and our RICE on all 34 categories. Our RICE consistently outperforms FashionNet on all categories, and outperforms SwinB on most of the categories. Due to RICE of averaging the features of frames as video features, temporal information is not effectively utilized. But SwinB introduces 3D shifted windows to preserve temporal dynamics. As a result, SwinB performs well on certain categories with occlusions or view changes, e.g., Suitcase&bag, as shown in the 5-th row of Fig. 3. The SwinB provides a promising way to enhance our model.

5.7. Attention Region Visualization

To provide insight into PMD, we conduct further visualization. In Fig. 8, we show the attention map of RICE\textsubscript{patch} between shop image and video patches, where an image is regarded as the query, and attention weights on all spatial patches are visualized. We use the attention weights in the cross-attention layer of PMD for visualization. We make the following observations. (1) For the complex scenarios like (a) in Fig. 8, our approach can distinguish the target Chinese liquor from the nearby background liquors. (2) Interestingly, as shown in (b) of Fig. 8, even the target product is not always visible in the video, our approach still focuses on the corresponding regions accurately while pays less attention to the occluded regions.
References


