Temporal-consistent CAMs for Weakly Supervised Video Segmentation in Waste Sorting

Andrea Marelli, Luca Magri, Federica Arrigoni, and Giacomo Boracchi

DEIB - Politecnico di Milano, Italy

Abstract. In industrial settings, weakly supervised (WS) methods are usually preferred over their fully supervised (FS) counterparts as they do not require costly manual annotations. Unfortunately, the segmentation masks obtained in the WS regime are typically poor in terms of accuracy. In this work, we present a WS method capable of producing accurate masks for semantic segmentation in case of video streams. More specifically, we build saliency maps that exploit the temporal coherence between consecutive frames in a video, promoting consistency when objects appear in different frames. We apply our method in a waste-sorting scenario, where we perform weakly supervised video segmentation (WSVS) by training an auxiliary classifier that distinguishes between videos recorded before and after a human operator, who manually removes specific wastes from a conveyor belt. The saliency maps of this classifier identify materials to be removed, and we modify the classifier training to minimize differences between the saliency map of a central frame and those in adjacent frames, after having compensated object displacement. Experiments on a real-world dataset demonstrate the benefits of integrating temporal coherence directly during the training phase of the classifier. Code and dataset are available upon request.

Keywords: Waste sorting · Weakly supervised video segmentation · Class activation maps

1 Introduction

With the escalation of global waste production, it has become critical to improve modern waste management systems. In particular, waste sorting involves the separation of specific types of recyclable waste, which usually consists in manually removing objects of different materials from a conveyor belt, where only a specific material must remain. Machine vision systems and deep learning have emerged as promising solutions to automatize these processes, aiming to enhance the efficiency and accuracy of waste management to reduce human error and to lower operational costs [9, 11]. These advancements significantly contribute to more sustainable and environmentally friendly practices. Unfortunately, Fully supervised (FS) segmentation methods, which are known for their efficacy in these tasks, require extensive pixel-level annotations for training. These annotations must be obtained by manually segmenting a large number of images, and they are extremely costly to produce.

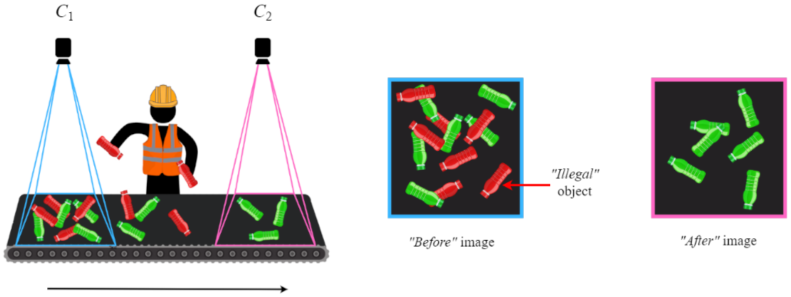


Fig. 1: Two cameras, C_1 and C_2, are placed along a conveyor belt where a human operator manually removes illegal objects. Camera C_1 captures the belt section before the operator's intervention, while Camera C_2 captures the section after, where only legal objects remain. Given a "before" image, our goal is to accurately segment objects into two categories: legal objects and illegal objects that should be removed.

In response to these challenges, we present a weakly supervised (WS) solution for waste sorting scenarios like the one illustrated in Fig. 1, that employs a dual-camera setup along a conveyor belt to streamline the recycling process. One camera captures images of the belt before any manual sorting, while the other captures images after unwanted items have been removed. The goal is to develop a method that automatically segments the objects in these images into two categories: the legal objects that should remain and the illegal ones that need to be removed. The idea proposed in Zerowaste [2], which presents a similar scenario, consists of training an auxiliary classifier to distinguish between "*before*" and "*after*" images. Since the "*before*" images are characterized by illegal objects that are not present in the "*after*" ones, the classifier learns to identify the illegal objects as a distinguishing element for "*before*" images. Once this binary classifier has been trained, saliency maps (CAMs) [44] can be used to locate illegal objects in "*before*" images: in this way, it is possible to obtain segmentation masks of illegal objects without the need for pixel-wise annotations. This is a general approach that can leverage any saliency map. Specifically, for the solution exposed in [2], authors use PuzzleCAM [17], which generates spatially consistent maps by dividing the image into smaller patches and ensuring consistent activation across these patches. This self-supervised segmentation approach used in [2] suffers from two main drawbacks: first, the generated classifier is biased towards the background of the images; secondly, the temporal correlation between saliency maps extracted from consecutive frames of the same camera is not taken into account.

We propose a WS solution that improves the one in [2] by exploiting both temporal and spatial coherence. Given a collection of videos acquired from both the *before* and *after* cameras, our novel deep-learning framework leverages both temporal and spatial coherence to generate accurate segmentation masks directly from the saliency maps. More specifically, while training the auxiliary classifier that provides saliency maps of illegal objects, we promote that the saliency maps of the same objects moving in different frames are similar, incorporating in the PuzzleCAM [17] process a novel reconstruction loss between the map of a central frame X_t and the aggregation of the motion-compensated maps of adjacent frames X_{t-1} and X_{t+1}. Specifically, to adjust the adjacent maps, we employ an optical flow algorithm [36], which computes the motion between consecutive frames in a video sequence, returning for each pixel both the direction and magnitude of movement. Our reconstruction loss forces the network to produce identical outputs for seemingly different but conceptually identical frames, allowing our method to accurately highlight and locate illegal objects over time. Consequently, our classifier is trained simultaneously to classify frames in before and after classes, and to provide accurate segmentation masks, ensuring that the network learns to recognize and segment objects based on their temporal dynamics as well as their appearance.

We are the first to leverage a reconstruction loss between saliency maps of nearby frames. To the best of our knowledge, no existing work utilizes this principle at a temporal level. PuzzleCAM [17] successfully employs this principle from a spatial perspective within a static image but not across multiple frames. Furthermore, to ensure the classifier focuses on the features of the objects rather than on the background, we separate the background from the images, formulating an auxiliary three-class classification problem, rather than the traditional binary classification problem considered in the literature. This results in segmentation masks comprising "*before*", "*after*", and "*background*" pixels.

Our experiments demonstrate that our approach provides segmentation masks that are very accurate and consistent over time, suitable for industrial waste sorting applications, distinguishing between legal and illegal objects without using any detailed pixel-level annotation. The paper is organized as follows. Section 2 reviews previous work, Section 3 formally defines the problem we address and Section 4 introduces our approach. Experiments are reported in Section 5 while Section 6 draws the conclusions.

2 Related Works

The landscape of segmentation methods based on neural networks can be organized into two main categories: *i*) FS approaches, that require pixel-level annotated datasets for precise segmentation, and *ii*) WS ones, that exploit image-level annotations and are definitely more practical for large-scale applications. In waste sorting, existing datasets also reflect this classification. After presenting an overview of these two categories, we will focus on WS methods that, as in our scenario, take as input videos instead of images.

Fully supervised image segmentation approaches span a wide range of methods, from simpler region-proposal [5, 8, 10, 12, 30] and fully convolutional networks [1, 6, 25, 43] to transformers [40]. The success of these methods heavily depends on the availability of precise pixel-level annotation for training. In waste sorting, this results in requiring a large training set $\mathbb{D} = \{(X_i, M_i)\}_{i=1}^{N}$ composed by RGB images X_i, each coupled with its segmentation mask M_{X_i} for different waste classes Λ, such as cardboard, metal, glass, and plastic, to name a few samples [2, 13, 20, 27, 28, 37].

While very useful for general waste sorting, these datasets have significant limitations in real-world applications since very rarely the segmentation task to be addressed in practical applications corresponds to those represented in these datasets. Therefore, one has to reset to manually label several images to train FS methods, which is expensive, time-consuming and challenging, making FS unfeasible especially in facilities that recycle specific materials.

Weakly Supervised image segmentation techniques remove the need of pixel-level annotations M on images by exploiting various forms of incomplete or imprecise supervision, such as bounding boxes [7,18,26,32], scribbles [24,34,35], and image-level class labels [14–16,19,22,31,33,39]. The latter include Class Activation Maps (CAMs) [44], which highlight regions of an image that are the most relevant to the class prediction made by a classifier. In this way it is possible to obtain coarse segmentation maps of a specific object using only a classifier trained at image-level. These segmentation masks can be considered as noisy pseudo-labels for training segmentation models [15, 19, 31, 33] to get more accurate segmentations.

In recent years, several extensions of CAMs have been developed. Grad-CAM [29], which utilizes the gradients of the target class flowing into the final convolutional layer to generate class-specific saliency maps, is perhaps the most popular solution. Another notable extension, and the one that most inspired this paper, is PuzzleCAM [17], which enforces spatial coherence among saliency maps of different patches that constitute the whole image.

In waste sorting, even if generally used for classification tasks, image-based waste datasets $\mathbb{D} = \{(X, y)_i\}_{i=1}^{N}$, such as `TrashNet` [41] and `TrashBox` [21], can be used to train WS segmentation networks due to their simple preparation process. However, the waste categories $y_i \in \Lambda$ (such as glass, paper, cardboard, plastic, metal, and general waste) in these datasets are too broad for industrial needs, which require more detailed distinctions between colored or transparent PET. Furthermore, most datasets are focused on waste images from domestic environments and are thus unsuitable for our industrial setting.

A notable exception is the ZeroWaste project [2], which, like our work, is collected in a recycling facility. The ZeroWaste dataset is divided into two parts: a widely used supervised component (ZeroWaste-f) and a largely unexplored unsupervised component (ZeroWaste-w). Similarly to our approach, the latter includes images collected "*before*" or "*after*" manual removal from a conveyor belt. Thus it is possible to utilize the WS solution described before. Although the ZeroWaste-w dataset has proposed the method of using saliency maps to identify illegal objects in the "before" images, this approach is only sketched and not fully developed in the literature. While we are inspired by this approach, our method extends and refines it. It is widely known that when we use saliency maps for image patches of the same class, the model focuses on key features and only identifies small discriminative parts of a target object [15, 38, 42]. To face this challenge and improve the performance, instead of considering static images, we take into account videos and enforce both spatial and temporal coherence. Therefore, our method belongs to the category of WSVS methods described below.

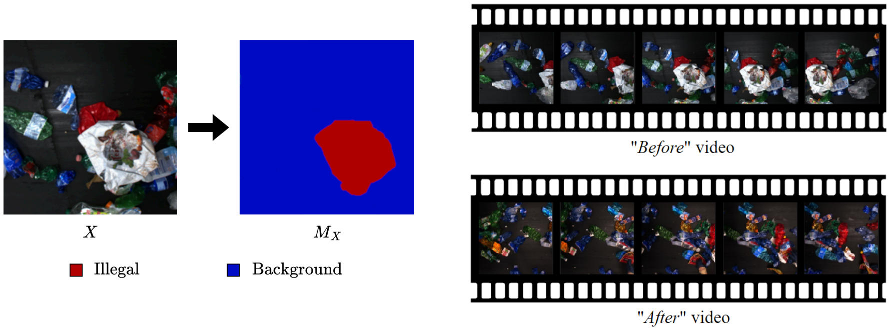
Another limitation of the ZeroWaste maps, which we address in our solution, is that images of the same class are collected under the same lighting conditions, resulting in a bias of the auxiliary classifier to recognize the class of an image based on the background characteristics rather than on the type of objects present in the image. Such a bias is automatically reflected in the segmentations obtained with saliency maps. In order to overcome this drawback, we take into account saliency maps both on the "before" and "after" categories, separating the background from the foreground and using it as a third class. It is also worth mentioning that ZeroWaste-w only leverages video data for the "before" class, while the "after" class consists of static images, and our method requires temporal coherence across both classes to be effective.

Weakly supervised video segmentation methods consider a whole video sequence $V = \{X_t\}_{t=1}^{T}$ annotated with a video label y, providing very easy-to-obtain annotation for the neural network. The main difference in using videos V rather than single images X consists in exploiting the rich temporal information available in videos. This allows for the propagation of information across frames, which can be exploited to enhance segmentation accuracy and coherence.

Since it is widely known that saliency maps focus on different zones of a single object, activating maps in different frames might highlight different parts of the same objects, due to the different displacement and lighting conditions. For this reason, temporal coherence has been exploited by several approaches to enhance segmentation or localization performance in videos.

Typically, a classifier is trained on static images to generate saliency maps for individual video frames. These maps are then combined to create comprehensive saliency maps used as supervision for a FS network. Frame-to-Frame (F2F) [23] uses optical flow to warp neighboring maps to a single frame, aggregating them in a post-processing phase to generate detailed maps for the FS network. T-CAM [3] employs a similar process but reuses the auxiliary classifier as an encoder for the segmentation network, overlapping neighboring maps without translation or optical flow compensation. CoLo-CAM [4] improves T-CAM by applying a color-based CRF filter on adjacent frames to ensure similar activations in regions with similar colors. In any case, both T-CAM and CoLo-CAM address the task of localization, which is not optimal for our scenario, given the strongly occluded nature of the images we are analyzing. In fact bounding boxes instead of segmentation masks would result in a lot of overlapping, leading to results that would be confused and of limited use. Furthermore, neither T-CAM nor CoLo-CAM uses optical flow. To the best of our knowledge, no existing architecture leverages the advantages of using temporal information during the classifier's training phase.

In contrast, we combine the principles of F2F [23] and PuzzleCAM [17] and train a classifier directly on videos, forcing it to generate precise and temporally



(a)

(b)

Fig. 2: Problem formulation: (a) an RGB input image X is processed to generate an accurate output mask M_X. This mask classifies each pixel as illegal (red), or background (blue).(b) Training set comprising "before" and "after" videos. "Before" videos capture the conveyor belt before human intervention. "After" videos capture the belt after non-colored PET objects have been removed.

consistent saliency maps by integrating spatial coherence from PuzzleCAM with temporal coherence from video data. This approach ensures that segmentation masks are as accurate as possible, with the temporal dimension incorporated from the initial training.

3 Problem formulation

We frame our waste sorting problem as a WS segmentation task where the training data is a set of images collected by the cameras C_1 and C_2 as shown in Figure 1. We refer to images collected before the human intervention as "*before*" images and to those collected after it as "*after*" images. Also, we refer to objects that the human operator must remove as "*illegal*" objects while to all the objects that must remain on the belt as "*legal*" objects. Given an RGB image $X \in \mathbb{R}^{w \times h \times 3}$ of the conveyor belt, with values normalized between $[0, 1]$, we aim at segmenting the *illegal* objects that the operator must remove. As illustrated in Figure 2a, this consist in estimating for the image X a *semantic segmentation mask* $M_X \in \Lambda^{w \times h}$ defined as:

$$\begin{aligned} M_X(r, c) = y \text{ if pixel at position } (r, c) \text{ in } X \text{ belongs to an} \\ \text{object of class } y \in \Lambda, \end{aligned} \tag{1}$$

where $\Lambda = \{0, 1\}$ is the set of *illegal* objects and *background* respectively. Note that in this formulation *legal* objects are segmented together with the *background.*

We make the following assumptions: the training set is composed by videos captured by cameras C_1 and C_2, which are labelled as "*before*" or "*after*" respectively. Thus, we are in a WS setting, *i.e.*, we only know which frames belong to the *before* category and which ones belong to the *after* category. Formally, the training set $\mathbb{D}$ is defined as follows: $\mathbb{D} = \{(V, \hat{y})_i\}_{i=1}^{N}$ denotes a set of N videos, where $V = \{X_t\}_{t=1}^{T}$ is an input video with T RGB frames X_t defined as above, and $\hat{y} \in \varLambda = \{0, 1\}$ is the class label representing to (*before*, *after*), that is only at video-level (Fig. 2b), where (*before*, *after*) videos directly correspond to the presence of (*illegal*, *background*) objects.

4 Proposed Solution

Inspired by `zerowaste-w` [2], we train an auxiliary classifier to distinguish between videos taken before and after human intervention. The classifier learns to identify the *before* video thanks to the presence of illegal objects that are instead absent in the *after* videos. A saliency map of each individual *before* frame would roughly highlight the regions corresponding to illegal objects, but the resulting segmentation masks might not be very accurate. Thus, to boost the accuracy of saliency maps, we exploit both the spatial and the temporal coherence of the videos, operating on triplets of consecutive frames X_{t-1}, X_t, X_{t+1} as outlined in Fig. 4. As a first step, we remove the background from the images of our dataset in a pre-processing step described in Section 4.1 (Fig. 3). As shown in Fig. 4, the background-removed frames are processed through a pre-trained backbone network (ResNet50) to extract features (Sec. 4.2), to be handled by two different modules. The **spatial module** (Sec. 4.3) implements the principles of Puzzle-CAM [17] and returns the reconstructed feature space f_t^{puzzle}, obtained by splitting the central frame X_t into local patches $X_t^{i,j}$ and by merging back their feature spaces $f_t^{i,j}$ computed on individual patches (Fig. 5). The **temporal module** (Sec. 4.4) operates along the temporal dimension of videos. It takes as input the adjacent frames X_{t-1} and X_{t+1} and, by exploiting optical flow, it reconciles the warped mask M_{t-1} and M_{t+1} and into a central, single, fused M_t^{fused} (Fig. 6). These two modules produce different outputs, which are then compared against the classifier's output with two reconstruction losses. This process forces the classifier to generate consistent saliency maps at spatial and temporal levels (Fig. 4).

4.1 Pre-processing

All "*before*" images share the same camera C_1, lighting conditions, and belt section, resulting in having all similar backgrounds. The same holds for "*after*" images. However, "*before*" and "*after*" backgrounds are very different from each other. Unfortunately, this condition results in the classifier focusing on the background instead of the features of the objects. For this reason, we preliminary segment foreground objects from the background in our dataset. For both *before* and *after* videos, we estimated a background by computing the pixel-wise median image across all grayscale frames. For each frame, the distance of every pixel with respect to the background estimator is then computed. Pixels significantly different from the estimator are so considered as foreground, resulting in a binary mask that is then applied to the RGB images.

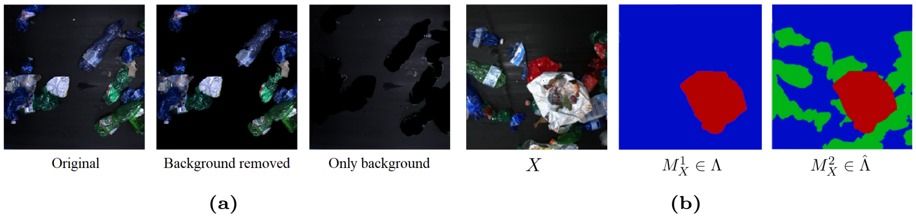


Fig. 3: (a) Comparison of images with background, without background and the extracted background itself, which is used to generate a third independent class, respectively. (b) By shifting from the Λ class domain to the $\hat{\Lambda}$ class domain, we can not only distinguish between *illegal* (red) and *background* (blue) elements, but also segment *legal* (green) objects with a new, more specific, label, distinguishing them from the empty belt regions.

Then, by inverting the binary masks, we generated a new set of images containing only background elements, expanding the dataset. This results in three classes of images: *after* without background, *before* without background, and only *background* images from both before and after sets. In this way, we drive the classifier to recognize as relevant only the features related to the objects, shifting from the set of classes Λ = {0,1}, corresponding to (*before*, *after*) to $\hat{\Lambda}$ = {0,1,2}, corresponding to (*before, after, background*), as shown in Figure 3a. Figure 3b illustrates that we use the extracted background itself as a distinct element from legal and illegal objects, which can now be both segmented with specific labels.

4.2 Feature Extraction and Classification Loss

As shown in Fig. 4, each frame X_t is processed through a pre-trained ResNet50 backbone F with a classification head θ that reduces the number of final feature maps to $|\hat{\Lambda}|$, corresponding to the three classes *after*, *before*, and *background*. The output of the backbone is the feature space f_t:

$$f_t = F(X_t). \tag{2}$$

which is then processed by a Global Average Pooling (GAP) layer G to produce the prediction vector $\hat{z} = \sigma(G(f_t))$ used for image classification. We utilize a multi-label soft margin loss for this task. For notational convenience, we define $\bar{z}$ as:

$$\bar{z} = \begin{cases} \hat{z}, & \text{if } z = 1 \\ 1 - \hat{z}, & \text{otherwise} \end{cases} \tag{3}$$

and define the classification loss as:

$$\text{cls}(\hat{z}, z) = -\log(\bar{z}) \tag{4}$$

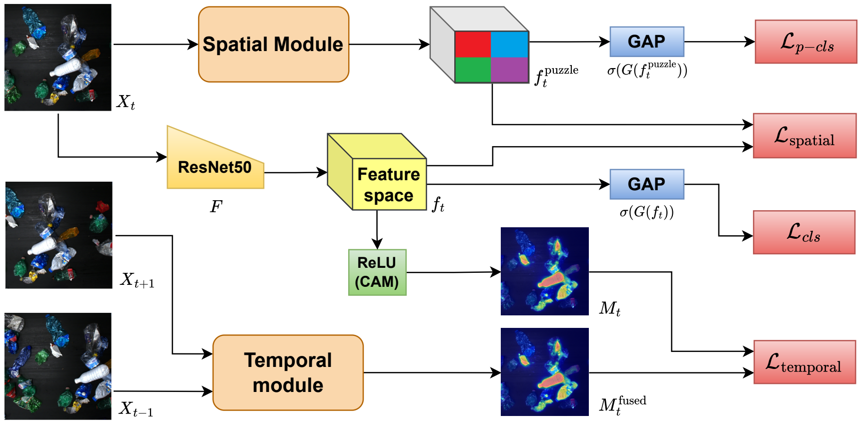


Fig. 4: Main pipeline illustration. The overall workflow of our network, which processes a triplet of frames (X_{t-1}, X_t, X_{t+1}). The spatial module (PuzzleCAM [17]) outputs a reconstructed feature space f_t^{puzzle} which is pushed to match the original feature space f_t by $\mathcal{L}_{\text{spatial}}$. The temporal module outputs a new saliency map M_t^{fused} for the central frame X_t, obtained from the features of the adjacent frames X_{t+1} and X_{t-1}. M_t^{fused} is then pushed to match the original map M_t by the reconstruction loss $\mathcal{L}_{\text{temporal}}$. $\mathcal{L}_{cls}$ and $\mathcal{L}_{p-cls}$ are instead the classification losses. The computation of the four losses of the network is described in Sec. 4.5, while spatial and temporal modules are detailed in Fig. 5 and 6, respectively.

where z is the true label vector of the image X_t corresponding to its class $y \in \Lambda$. The classification loss for X_t is then computed as:

$$\mathcal{L}_{cls} = \text{cls}(\hat{z}, z) \tag{5}$$

which is used to train the classifier for the image classification task.

4.3 Spatial Module

Following PuzzleCAM [17], our architecture (Fig. 5) is designed to promote spatial coherence of saliency maps when extracting features from a single image as follows. The Spatial Module processes the central frame X_t to match its features with those extracted from its patches. More specifically, from an input image X_t of size $w \times h$, the tiling module generates non-overlapping tiled patches $\{X_t^{1,1}, X_t^{1,2}, X_t^{2,1}, X_t^{2,2}\}$ of size $\frac{w}{2} \times \frac{h}{2}$. Next, we extract $f_t^{i,j}$ feature spaces for each $X_t^{i,j}$ as described in (2). Finally, the merging module assembles all $f_t^{i,j}$ into a single feature space f_t^{puzzle} that has the same shape as f_t, the feature space of the original image X_t (Fig. 5). Using the GAP layer G described in Section 4.2, we map f_t^{puzzle} into a prediction vector $\hat{z}^{\text{puzzle}} = G(f_t^{\text{puzzle}})$. Using (3) and (4) we compute a new classification loss as

$$\mathcal{L}_{p-cls} = \text{cls}(\hat{z}^{\text{puzzle}}, z), \tag{6}$$

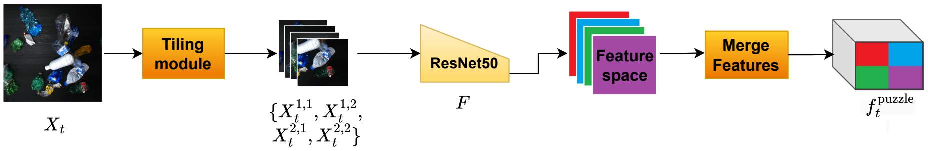


Fig. 5: Spatial Module: The central frame X_t is divided into non-overlapping patches by the tiling module, and for each patch, we extract its feature maps. These sub-feature maps are then re-merged to create a single reconstructed feature space that is compared with the one of the original image X_t through the reconstruction loss $\mathcal{L}_{\text{spatial}}$. This module is the implementation of PuzzleCAM and it aims to improve segmentation by focusing on the spatial arrangement of objects within a single frame.

that improves the image classification performance. To ensure the classifier produces spatially consistent CAMs, we incorporate a reconstruction loss, which aligns the original and reconstructed feature spaces. This loss is defined as:

$$\mathcal{L}_{\text{spatial}} = \|f_t - f_t^{\text{puzzle}}\|_1. \tag{7}$$

4.4 Temporal Module

The main contribution introduced by our work is the temporal module, through which we compute temporal consistent saliency maps: this module processes a triplet of frames X_{t-1}, X_t, and X_{t+1} in a joint classification network employing temporal coherence between the saliency maps of the frames (Fig. 6).

CAM generation. First, from X_{t-1} and X_{t+1}, we extract feature spaces f_{t-1} and f_{t+1} as in (2). Then, for every frame of the triplet X_{t-1}, X_t, X_{t+1}, as done in [17], we use a ReLU activation function to compute the saliency map M for the class y the input images belong to:

$$M = \text{ReLU}(f[y]), \tag{8}$$

where $f[y]$ represent the y-th channel of a feature space f. The computed map M is then normalized by dividing it by its maximum value. We produce the saliency maps for every frame in triplet X_{t-1}, X_t, and X_{t+1} obtaining M_{t-1}, M_t, and M_{t+1}, respectively.

Optical Flow Warping and CAM Fusion. As next step, we align the saliency maps using DICL-FLow [36], and in practice we compute the optical flows between consecutive frames X_t and $X_{t\pm1}$ and use these flows to warp the lateral maps to the central one:

$$\begin{gathered} M_i^{\text{warped}} = \text{Warp}(M_i,\ \text{Flow}(X_t, X_i)) \\ \text{for } i \in \{t-1,\ t+1\}, \end{gathered} \tag{9}$$

where $\text{Flow}(X_t, X_i)$ denotes the optical flow between frame X_t and frame X_i. The warped maps for frames X_{t-1} and X_{t+1} are then fused keeping the pixel-wise maximum:

$$M_t^{\text{fused}} = \max(M_{t-1}^{\text{warped}}, M_{t+1}^{\text{warped}}) \tag{10}$$

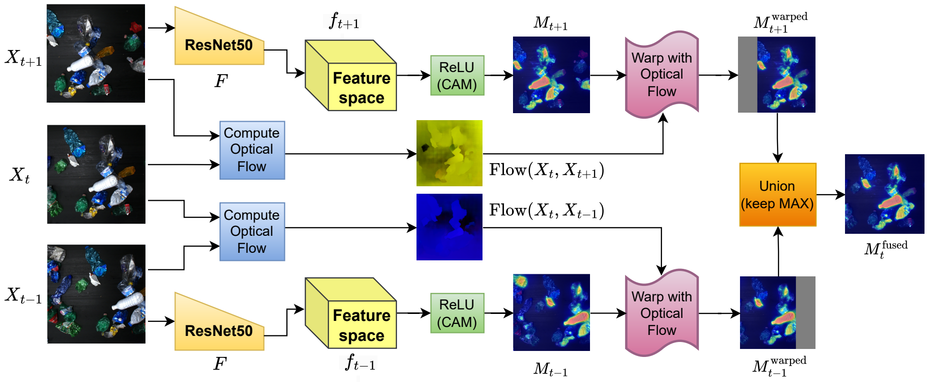


Fig. 6: Temporal module: It processes two frames X_{t-1} and X_{t+1} adjacent to X_t, to extract their saliency maps. These maps are then warped using optical flow to align them temporally and fused in M_t^{fused} keeping the pixel-wise maximum values. M_t^{fused} is then compared with the map of the central X_t through the reconstruction loss $\mathcal{L}_{\text{temporal}}$. This process ensures that the activations of objects are temporally coherent, promoting the network to segment objects consistently across frames.

In this way, we obtain a new saliency map for the central frame X_t based on the activation of the features identified in the lateral frames. Figure 6 illustrates the computation of optical flow, warping of the maps and union of them in blue, purple and dark orange modules, respectively.

Temporal Module Losses. Using the same principle as PuzzleCAM, we add a reconstruction loss to force the original saliency map M_t of central frame X_t to be closed of the reconstructed one from the adjacent frames M_t^{fused}, namely:

$$\mathcal{L}_{\text{temporal}} = \|M_t - M_t^{\text{fused}}\|_1. \tag{11}$$

In this way, the activations of an object in different positions are temporally coherent.

4.5 Final Loss Design

To summarize, as illustrated in Fig. 4, we train our network by minimizing a loss function that combines the losses from both the Spatial (PuzzleCAM) and the Temporal modules. The final loss function $\mathcal{L}_{\text{total}}$ is the sum of the classification losses given by Eq. (5) and (6), and the reconstruction losses given by (7) and (11), namely:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{cls} + \mathcal{L}_{p-cls} + \alpha\mathcal{L}_{\text{spatial}} + \beta\mathcal{L}_{\text{temporal}}. \tag{12}$$

where α and β are regularization terms that weight respectively the spatial and temporal coherence components given.

Table 1: Comparison of mIoU scores for different methods across the ZEROWASTE and SERUSO datasets. The superior performance of Frame-2-Frame over GradCAM demonstrates how the benefit is evident even using temporal coherence only in the post-processing phase. On the SERUSO dataset, our model, which incorporates both temporal and spatial coherence during training, outperforms all other CAM-based methods, including PuzzleCAM.

IoU	**GradCAM**	**Frame-2-Frame**	**PuzzleCAM**	**Our**
SERUSO	22.08	27.94	34.20	37.84
ZEROWASTE	23.13	26.43	29.87	Impossible

5 Experiments

This section is devoted at assessing the benefits of our solution on both segmentation and classification tasks. After describing our dataset, we present a comparative analysis of our approach against baseline methods, demonstrating the effectiveness of exploiting both temporal coherence in segmentation and background removal in classification tasks.

5.1 Datasets and Competitors

We evaluate our method on our custom-collected dataset (named SERUSO and available upon request), which consists of 3682 images, divided into 36 videos for the "after" class and 32 videos for the "before" class. Specifically, there are 1836 "after" images and 1846 "before" images, each with a resolution of 2400 × 2400 pixels. Cameras have been installed to monitor a conveyor belt containing objects made from PET materials, including transparent, bluish and opaque PET. The operators remove any object but semi-transparent colored PET ones. As a result, the "before" images captured the initial, mixed material flow, while the "after" images contain primarily semi-transparent colored PET objects with occasional anomalies (see Fig. 7 for an example). A total of 364 images were manually labeled by segmenting "illegal" objects in the "before" images. These segmentation masks were used exclusively for testing. We also performed additional experiments on the Zerowaste-w dataset [2], extending the range of analyzed methods beyond those used by the authors. We benchmarked our method against the approach used by the authors of Zerowaste-w, namely PuzzleCAM [17], as well as other CAM-based methods, including Grad-CAM and its extension incorporating temporal coherence (Frame-to-frame [23]). In addition, we conducted ablation studies to assess the impact of each component of our method.

5.2 Results

All experiments were conducted on a workstation equipped with an Nvidia RTX A6000 GPU. The images were re-scaled to 512 × 512 as the network inputs and the dataset was split into training and validation sets with an 80% and 20%

Table 2: IoU scores of models trained with different reconstruction loss configurations—none, only temporal, only spatial, and both—show significant performance improvements with spatial and temporal coherence. On the SERUSO dataset, the temporal module alone greatly enhances performance compared to no module but falls short of the spatial-only module (PuzzleCAM). While the spatial module outperforms the temporal module, combining both yields the highest performance, demonstrating their complementary benefits.

IoU	None	Only Temporal	Only Spatial	Both
SERUSO	24.08	29.23	34.20	37.84

split, respectively. In all experiments, α and β are set to 0 for the first epoch and then linearly increased to a maximum of 4 by the midpoint of training, gradually prioritizing reconstruction losses over classification losses.

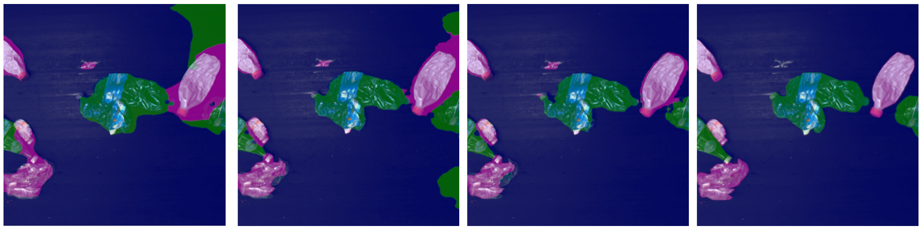
Segmentation. To assess the segmentation performance of our saliency maps, we computed the mean Intersection-over-Union (mIoU) over the "before" class. Saliency map-based methods identify the most relevant regions for a classifier to assign an image to a specific class. Therefore, all experiments focused on segmenting *illegal* objects in "*before*" images. While confirming that advanced methods like PuzzleCAM show substantial improvements over traditional techniques like GradCAM, Tab. 1 also demonstrates how techniques utilizing temporal coherence significantly enhance segmentation performance. In particular, our method outperforms all others on the SERUSO dataset, showcasing the superiority of integrating temporal coherence directly in the training phase for the segmentation task. Figure 7 shows an example of the qualitative results obtained on SERUSO dataset, highlighting the differences in segmentation performance among various methods.

Unfortunately, it is impossible to train our method on Zerowaste since it provides only static data for the "after" class, preventing the incorporation of temporal coherence in training. Table 2 shows how our method performs on the various modules when considered individually. This ablation study demonstrates that both the Spatial and Temporal modules alone outperform a simple saliency map from a classifier trained with only classification losses, whereas their combined use surpasses both the individual configurations.

Classification. In order to assess the impact of background removal, we evaluate the classification accuracy of a standard classifier (ResNet 50) on two versions of the Zerowaste-w dataset, one with the original images and one with images having the background removed, as explained in Section 4.1. Results, in Tab. 3, show that no background-trained classifier performs well on both the scenarios (with and without background), while the classifier trained on the datasets with background performs well on its training set, but badly on the other, demonstrating the bias given by the background.

Table 3: Classifier accuracy on the Zerowaste-w dataset w and w/o backgrounds. The classifier trained on data with backgrounds performs excellently when tested on a dataset with backgrounds, but its performance drops when tested on a dataset w/o backgrounds. Conversely, the classifier trained on data without backgrounds achieves similar high performance when tested on both datasets with and w/o backgrounds.

CLASSIFIER →	**BACKGROUNDS**		**NO BACKGROUNDS**	
DATASET ↓	Train	Val	Train	Val
BACKGROUNDS	100	99.69	98.25	96.12
NO BACKGROUNDS	68.97	64.62	99.57	98.02



(a) GradCAM **(b)** PuzzleCAM **(c)** Ours **(d)** Ground Truth

Fig. 7: Qualitative comparison of segmentation results on the SERUSO dataset, showing how our method (c) attains more precise segmentation compared to both GradCAM (a) and PuzzleCAM (b) thanks to the combined exploitation of spatial and temporal coherence.

6 Conclusions

We addressed the challenging task of industrial waste sorting using a WSVS approach. We proposed a novel method that drives a classifier to produce temporal consistent saliency maps for objects appearing in different frames. Experiments and ablation studies demonstrated that the use of temporal coherence directly in the classifier's training phase effectively improves the classifier's ability to generate saliency maps, outperforming the mIoU of other saliency maps-based methods. The results obtained in our dual-camera setup are very promising and suggest that this approach can be applied to other industrial processes with similar settings, where it is necessary to manually separate specific objects from a heterogeneous stream e.g. in product quality control processes, where anomalous elements need to be removed from a stream of objects, such as damaged or faulty products. Given that saliency maps are currently computed during the inference phase using only a single frame, future work explores including adjacent frames in the map computation at inference time as well. Also, as a next step, the segmentation masks obtained can be used as pseudo-labels to supervise a FS segmentation network, to improve the segmentation performance.

Acknowledgements: This paper is supported by PNRR-PE-AI FAIR project funded by the NextGeneration EU program and by GEOPRIDE ID: 2022245ZYB, CUP: D53D23008370001 (PRIN 2022 M4.C2.1.1 Investment).

References

1. Adeyinka, A.A., Adebiyi, M.O., Akande, N.O., Ogundokun, R.O., Kayode, A.A., Oladele, T.O.: A deep convolutional encoder-decoder architecture for retinal blood vessels segmentation. In: Computational Science and Its Applications–ICCSA 2019: 19th International Conference, Saint Petersburg, Russia, July 1–4, 2019, Proceedings, Part V 19. pp. 180–189. Springer (2019)
2. Bashkirova, D., Abdelfattah, M., Zhu, Z., Akl, J., Alladkani, F., Hu, P., Ablavsky, V., Calli, B., Bargal, S.A., Saenko, K.: Zerowaste dataset: Towards deformable object segmentation in cluttered scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21147–21157 (2022)
3. Belharbi, S., Ben Ayed, I., McCaffrey, L., Granger, E.: Tcam: Temporal class activation maps for object localization in weakly-labeled unconstrained videos. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 137–146 (2023)
4. Belharbi, S., Murtaza, S., Pedersoli, M., Ayed, I.B., McCaffrey, L., Granger, E.: Colo-cam: Class activation mapping for object co-localization in weakly-labeled unconstrained videos. arXiv preprint arXiv:2303.09044 (2023)
5. Caesar, H., Uijlings, J., Ferrari, V.: Region-based semantic segmentation with end-to-end training. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14. pp. 381–397. Springer (2016)
6. Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proceedings of the European conference on computer vision (ECCV). pp. 801–818 (2018)
7. Dai, J., He, K., Sun, J.: Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In: Proceedings of the IEEE international conference on computer vision. pp. 1635–1643 (2015)
8. Dai, J., He, K., Sun, J.: Convolutional feature masking for joint object and stuff segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3992–4000 (2015)
9. Fan, J., Cui, L., Fei, S.: Waste detection system based on data augmentation and yolo ec. Sensors **23**(7) (2023). https://doi.org/10.3390/s23073646, https://www.mdpi.com/1424-8220/23/7/3646
10. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 580–587 (2014)
11. Gundupalli, S.P., Hait, S., Thakur, A.: A review on automated sorting of source-separated municipal solid waste for recycling. Waste Management **60**, 56–74 (2017). https://doi.org/https://doi.org/10.1016/j.wasman.2016.09.015, https://www.sciencedirect.com/science/article/pii/S0956053X16305189, special Thematic Issue: Urban Mining and Circular Economy
12. Hariharan, B., Arbeláez, P., Girshick, R., Malik, J.: Simultaneous detection and segmentation. In: Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VII 13. pp. 297–312. Springer (2014)
13. Hong, J., Fulton, M.S., Sattar, J.: Trashcan 1.0 an instance-segmentation labeled dataset of trash observations (2020)
14. Hou, Q., Jiang, P., Wei, Y., Cheng, M.M.: Self-erasing network for integral object attention. Advances in neural information processing systems **31** (2018)

15. Huang, Z., Wang, X., Wang, J., Liu, W., Wang, J.: Weakly-supervised semantic segmentation network with deep seeded region growing. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7014–7023 (2018)
16. Jin, B., Ortiz Segovia, M.V., Susstrunk, S.: Webly supervised semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3626–3635 (2017)
17. Jo, S., Yu, I.J.: Puzzle-cam: Improved localization via matching partial and full features. In: 2021 IEEE international conference on image processing (ICIP). pp. 639–643. IEEE (2021)
18. Khoreva, A., Benenson, R., Hosang, J., Hein, M., Schiele, B.: Simple does it: Weakly supervised instance and semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 876–885 (2017)
19. Kolesnikov, A., Lampert, C.H.: Seed, expand and constrain: Three principles for weakly-supervised image segmentation. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14. pp. 695–711. Springer (2016)
20. Koskinopoulou, M., Raptopoulos, F., Papadopoulos, G., Mavrakis, N., Maniadakis, M.: Robotic waste sorting technology: Toward a vision-based categorization system for the industrial robotic separation of recyclable waste. IEEE Robotics and Automation Magazine **28**(2), 50–60 (2021). `https://doi.org/10.1109/MRA.2021.3066040`
21. Kumsetty, N.V., Nekkare, A.B., Kamath, S., et al.: Trashbox: trash detection and classification using quantum transfer learning. In: 2022 31st Conference of Open Innovations Association (FRUCT). pp. 125–130. IEEE (2022)
22. Lee, J., Kim, E., Lee, S., Lee, J., Yoon, S.: Ficklenet: Weakly and semi-supervised semantic image segmentation using stochastic inference. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5267–5276 (2019)
23. Lee, J., Kim, E., Lee, S., Lee, J., Yoon, S.: Frame-to-frame aggregation of active regions in web videos for weakly supervised semantic segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6808–6818 (2019)
24. Lin, D., Dai, J., Jia, J., He, K., Sun, J.: Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3159–3167 (2016)
25. Noh, H., Hong, S., Han, B.: Learning deconvolution network for semantic segmentation. In: Proceedings of the IEEE international conference on computer vision. pp. 1520–1528 (2015)
26. Papandreou, G., Chen, L.C., Murphy, K.P., Yuille, A.L.: Weakly-and semi-supervised learning of a deep convolutional network for semantic image segmentation. In: Proceedings of the IEEE international conference on computer vision. pp. 1742–1750 (2015)
27. Proença, P.F., Simoes, P.: Taco: Trash annotations in context for litter detection. arXiv preprint arXiv:2003.06975 (2020)
28. Sánchez-Ferrer, A., Gallego, A.J., Valero-Mas, J.J., Calvo-Zaragoza, J.: The cleansea set: a benchmark corpus for underwater debris detection and recognition. In: Iberian Conference on Pattern Recognition and Image Analysis. pp. 616–628. Springer (2022)
29. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: Visual explanations from deep networks via gradient-based localization. In:

Proceedings of the IEEE international conference on computer vision. pp. 618–626 (2017)

30. Shen, D., Ji, Y., Li, P., Wang, Y., Lin, D.: Ranet: Region attention network for semantic segmentation. Advances in Neural Information Processing Systems **33**, 13927–13938 (2020)
31. Shimoda, W., Yanai, K.: Self-supervised difference detection for weakly-supervised semantic segmentation. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 5208–5217 (2019)
32. Song, C., Huang, Y., Ouyang, W., Wang, L.: Box-driven class-wise region masking and filling rate guided loss for weakly supervised semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3136–3145 (2019)
33. Sun, G., Wang, W., Dai, J., Van Gool, L.: Mining cross-image semantics for weakly supervised semantic segmentation. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16. pp. 347–365. Springer (2020)
34. Tang, M., Djelouah, A., Perazzi, F., Boykov, Y., Schroers, C.: Normalized cut loss for weakly-supervised cnn segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1818–1827 (2018)
35. Vernaza, P., Chandraker, M.: Learning random-walk label propagation for weakly-supervised semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7158–7166 (2017)
36. Wang, J., Zhong, Y., Dai, Y., Zhang, K., Ji, P., Li, H.: Displacement-invariant matching cost learning for accurate optical flow estimation. Advances in Neural Information Processing Systems **33**, 15220–15231 (2020)
37. Wang, T., Cai, Y., Liang, L., Ye, D.: A multi-level approach to waste object segmentation. Sensors **20**(14), 3816 (2020)
38. Wei, Y., Feng, J., Liang, X., Cheng, M.M., Zhao, Y., Yan, S.: Object region mining with adversarial erasing: A simple classification to semantic segmentation approach. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1568–1576 (2017)
39. Wei, Y., Xiao, H., Shi, H., Jie, Z., Feng, J., Huang, T.S.: Revisiting dilated convolution: A simple approach for weakly-and semi-supervised semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7268–7277 (2018)
40. Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J.M., Luo, P.: Segformer: Simple and efficient design for semantic segmentation with transformers. Advances in neural information processing systems **34**, 12077–12090 (2021)
41. Yang, M., Thung, G.: Classification of trash for recyclability status. CS229 project report **2016**(1), 3 (2016)
42. Zhang, X., Wei, Y., Feng, J., Yang, Y., Huang, T.S.: Adversarial complementary learning for weakly supervised object localization. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1325–1334 (2018)
43. Zhang, Z., Zhang, X., Peng, C., Xue, X., Sun, J.: Exfuse: Enhancing feature fusion for semantic segmentation. In: Proceedings of the European conference on computer vision (ECCV). pp. 269–284 (2018)
44. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2921–2929 (2016)