WS^2: Weakly Supervised Segmentation using Before-After Supervision in Waste Sorting

Andrea Marelli[1] Alberto Foresti[2] Leonardo Pesce[1] Giacomo Boracchi[1] Mario Grosso[1]
[1] Politecnico di Milano, Italy [2] EURECOM, France
andrea.marelli@polimi.it, alberto.foresti@eurecom.fr, leonardo.pesce@mail.polimi.it,
giacomo.boracchi@polimi.it, mario.grosso@polimi.it

Abstract

In industrial quality control, to visually recognize unwanted items within a moving heterogeneous stream, human operators are often still indispensable. Waste-sorting stands as a significant example, where operators on multiple conveyor belts manually remove unwanted objects to select specific materials. To automate this recognition problem, computer vision systems offer great potential in accurately identifying and segmenting unwanted items in such settings. Unfortunately, considering the multitude and the variety of sorting tasks, fully supervised approaches are not a viable option to address this challange, as they require extensive labeling efforts. Surprisingly, weakly supervised alternatives that leverage the implicit supervision naturally provided by the operator in his removal action are relatively unexplored. In this paper, we define the concept of Before-After Supervision, illustrating how to train a segmentation network by leveraging only the visual differences between images acquired before and after the operator. To promote research in this direction, we introduce WS^2 (Weakly Supervised segmentation for Waste-Sorting), the first multi-view dataset consisting of more than 11 000 high-resolution video frames captured on top of a conveyor belt, including "before" and "after" images. We also present a robust end-to-end pipeline, used to benchmark several state-of-the-art weakly supervised segmentation methods on WS^2. [1]

1. Introduction

Despite the increasing automation in multiple industrial sectors, Human Operators (HOs) are still strictly necessary for numerous tasks. Many quality control processes, ranging from food to pharmaceutical lines, rely on HOs meticulously monitoring a continuous flow of items, identifying and manually removing anomalies or imperfections based on each item's visual appearance. This principle is reflected in modern recycling plants. Here, waste is first coarsely sorted by NIR-based automatic machines, then ends up on multiple separate conveyor belts where HOs manually remove undesired items from the stream, leaving on each dedicated belt only the objects of a specific material based on visual cues. Figure 1 illustrates this process: the HO discards the *unwanted* objects, leaving on the belt only the *wanted* ones. Supporting this activity with automatic tools would improve both the efficacy and efficiency of sorting plants, reducing the risks of injuries and alleviating stressful working conditions for HOs [24]. With the advent of deep learning, computer vision systems have emerged as promising solutions to automating sorting tasks. In fact, a semantic segmentation network can be trained to analyze images or

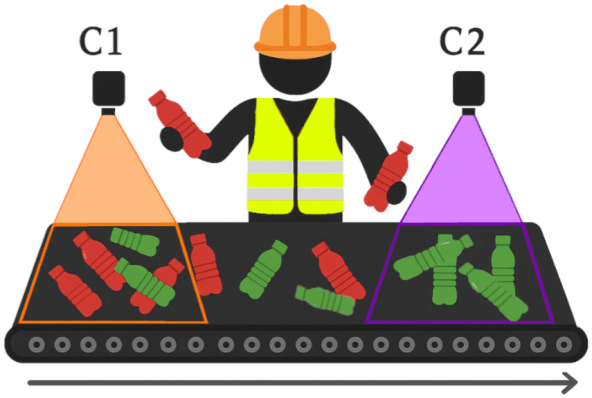
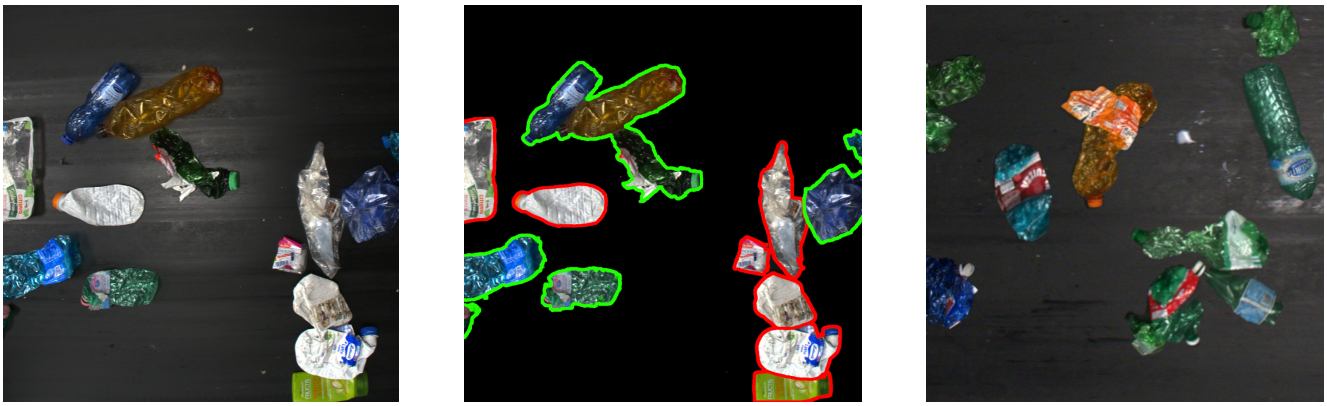


Figure 1. In a waste sorting plant, two cameras, C_1 and C_2, are placed along a conveyor belt where a human operator manually removes *unwanted* objects (red) from a mixed waste stream, leaving on the belt only *wanted* ones (green). C_1 captures the belt section *before* the HO's intervention, while C_2 captures the section *after*, where only *wanted* objects remain. We aim to train, without any pixel-wise annotation nor additional supervision, a binary segmentation model for *wanted*/*unwanted* objects.

[1] The WS^2 dataset is publicly available for download at https://zenodo.org/records/14793518, all the details are reported in the supplementary material.



(a) (b) (c)

Figure 2. (a) "*Before*" frame showing both *wanted* and *unwanted* waste items on the conveyor belt. (b) Annotation overlay: green outlines mark the *wanted* items to keep (semi-transparent PET bottles), while red outlines indicate *unwanted* waste to be removed (all other plastic materials). (c) "*After*" frame in which only the *wanted* items remain. The absence of the *unwanted* waste, present only in the "*before*" frames, serves as supervision in our WS segmentation approach, enabling the training of a network able to segment the *unwanted* items.

videos capturing the stream running on a belt and segment the *unwanted* items, supporting humans in their operation and paving the way for the system's automation.

In spite of substantial research on industrial waste sorting segmentation, existing efforts have been mainly based on datasets designed for Fully Supervised (FS) learning (Table 1). Unfortunately, gathering extensive manual annotations in the form of pixel-level annotated segmentation masks can be very time-consuming. Considering the fine-grained sorting demands of modern recycling plants, a dedicated detection model would be required for each specific line of work. Therefore, training each specific model in a FS fashion would require a large amount of data to be annotated for every belt, resulting in prohibitive costs.

In this paper, we investigate an alternative learning paradigm for segmenting *unwanted* items in sorting tasks, without the need to manually annotate the large quantity of ground-truth masks required to train a distinct segmentation network for each individual line. Our key insight lies in the fact that, comparing images of the belt sections captured *before* and *after* the HO's removal operation (Figure 2), it is possible to learn the supervisory signals for the segmentation task that is already implicit in the HO's activity. In fact, by removing the *unwanted* objects from the stream, the operator effectively indicates which items should be segmented without any manual annotations. We define this special supervision signal as "*Before-After*" supervision.

Nonetheless, considering that the HO's operation alters the relative position also of the *wanted* elements on the belt, straightforward solutions like subtracting (or directly comparing) pairs of corresponding *before* and *after* images is not effective to recover the desired segmentation masks. Thus, inspired by Zerowaste [2], we detailed a self-supervised learning solution in which an auxiliary classifier $\mathcal{K}_\theta$ is trained to distinguish between *before* and *after* images, and Saliency Maps (SMs) of $\mathcal{K}_\theta$ are then computed to leverage the implicit supervision provided by the HO. Specifically, the SMs for the *before* class on the corresponding images highlight the most representative regions of the *before* class, which essentially correspond to the unwanted objects, as these are never or rarely seen in the *after* images. Training a neural network to identify *unwanted* items via this Weakly Supervised (WS) framework would enable a vision system to automatically assess the HO's performance and provide corrective feedback, effectively learning the selection task performed by the HO directly from a video stream. This novel self-supervised paradigm not only learns a specific sorting criterion but, for each new conveyor belt, it requires only collections of before/after intervention frames, with no need for manual annotations.

To boost research in this area, our first contribution is WS^2 (Weakly Supervised segmentation for Waste Sorting), a comprehensive, open-access "Before-After" dataset collected in Seruso S.p.A., a real-world plastic sorting facility in Verderio (LC), Italy. Specifically, the WS^2 dataset comprises a sequence of more than 11 000 video frames acquired *before* and *after* a HO selecting semi-transparent colored PET objects from a conveyor line where only plastic items are present (Figure 2). This corresponds to a valuable benchmark in a waste sorting modern scenario. Unlike existing datasets (see Section 2.2), we specifically collected a large number of video frames to test WS segmentation methods exploiting temporal correlation in videos like [14].

In our second contribution, we design a comprehensive pipeline for general SM-based WS segmentation methods to exploit the *Before-After* supervision paradigm. The

pipeline, detailed in Section 3, encompasses SM computation, map refinement, and pseudo-mask generation to finally train a FS segmentation network without any form of human supervision. We implement our pipeline using mainstream SM methods, and we benchmark all these solution on WS^2.

Finally, we introduce a novel three-class training strategy for the auxiliary classifier $\mathcal{K}_\theta$ based on Background Removal (BR) (see Section 4). Since all images within the same class share a similar background, the classifier may inadvertently rely on background cues rather than object-specific features. To mitigate this bias, we explicitly model the background as a separate class alongside *before* and *after*, guiding the network to focus on object-level differences rather than background similarities and enhancing its ability to localize *unwanted* items. Our experiments demonstrate the potential for deriving generalizable WS segmentation solutions to enable the development of deep learning-based automation for manual sorting and quality control activities.

The "Before-After" training strategy we consider was initially briefly introduced in ZeroWaste [2], which focuses on high-level material segmentation (e.g., glass, cardboard, plastics) using a FS annotated dataset. ZeroWaste does not detail a pipeline to learn from the *Before-After* supervision and the *Before/After* data comprises a limited set of 1,400 images to separate only white paper from a mixed-material stream. This coarse formulation overlooks the finer-grained requirements of modern waste sorting scenarios, where distinctions between visually similar materials (e.g., different types of plastic) are often critical, even if humanly recognizable. Moreover, ZeroWaste does not account for background bias or temporal continuity between frames. In contrast, we show that leveraging both temporal coherence and a BR-based training strategy leads to significantly improved accuracy in WS video segmentation.

The paper is structured as follows. Section 2 reviews related work. Section 3 introduces the *Before-After* training pipeline, while section 4 focuses on the BR three-class training strategy we prepare. Section 5 presents the key characteristics and collection methodology of our dataset. Section 6 reports our experimental results, and Section 7 concludes the paper and outlines future directions.

2. Related Work

We discuss in this section the deep learning solutions previously implemented for both human activity understanding and waste sorting management. Section 2.1 explores related work on human activity understandig in similar contexts, while Section 2.2 illustrates existing waste datasets in the literature.

2.1. Human Activity Understanding

Our work connects to the broader field of skilled human–action understanding, where scarce annotations often motivate "learning by observing" approaches. A substantial thread of research investigates procedural operations in instructional videos, aiming to recognize complex steps without dense annotation. Cross-task weak supervision [31], for instance, leverages text-video alignments to transfer knowledge across procedures, while [13] uses narrated scripts to guide procedural activity recognition in long untrimmed clips. Egocentric video task translation [26] demonstrates how heterogeneous video understanding tasks can be unified by mapping outputs between multiple objectives, and [8] benchmarks comprehension of human tasks through question–answering over egocentric recordings. Although these methods excel at classifying or translating action sequences, they do not aim to train a model in reproducing or executing observed operations.

Another line of work focuses on procedural action recognition in industrial-like settings. Several egocentric and exocentric datasets capture two-handed assembly [1], anomalous manufacturing processes [15], and human–object interactions in industrial environments [18]. More recently, [20] introduces procedure step recognition with explicit modelling of execution errors in egocentric videos. While these benchmarks richly annotate object states and hand poses, they rely on full supervision and do not directly learn action dynamics purely by observing environmental changes. Finally, human-in-the-loop feedback has been applied to anomaly detection in illegal timber trade [5], where HOs iteratively label incoming data and progressively refine an anomaly score model, in an online learning process.

In our work, we introduce a novel framework to learn not from direct observation of the human actor but from the environment they alter: by comparing the scene *before* and *after* each intervention, we automatically extract, segment, and understand the performed environment modification. To explore this learning framework in a real-world scenario, we collected our WS^2 dataset in a waste sorting facility. To our knowledge, no prior work has primarily focused on such a self-supervised learning framework.

2.2. Waste Images Datasets

Waste identification datasets can be categorized based on their intended tasks: classification, object detection, and segmentation. Classification datasets [12, 22, 27] consist of images labeled with a single image-level category (e.g., "plastic," "glass"), while segmentation [2, 7, 11, 17] and object detection [6, 16] datasets provide finer pixel-level annotations. Beyond annotation type, waste datasets differ in the environment they were collected from: in the wild (outdoor, on a floor...) [6, 7, 12, 16, 17, 22], or in controlled laboratory settings [11, 27]. Only a few datasets [2, 28] are collected in real waste sorting plants. Table 1 summarizes this proprierties, comparing each dataset with ours. As can be seen from the table, there is a growing interest in both industrial and

Dataset	Images	Task	Environment	Supervision	Labels	Labels type	Video
TrashNet[27] (2016)	2 400	Classification	Laboratory	Full	6	Materials	No
LWW [22] (2019)	1 402	Classification	In the wild	Full	19	Materials	No
TACO[17] (2020)	1 500	**Segmentation**	Laboratory	Full	28	Materials	No
AquaTrash[16] (2020)	369	Detection	In the wild	Full	4	Materials	No
TrashCan[7] (2020)	7 212	Det./**Segm.**	In the wild	Full	16	Materials	**Yes**
ReSortIT[11] (2021)	21 600	**Segmentation**	Laboratory	Full	4	Materials	No
Trashbox[12] (2022)	17 785	Classification	In the wild	Full	20	Materials	No
ZeroWaste-f [2] (2022)	12 125	**Segmentation**	**Industrial**	Full	4	Materials	No
ZeroWaste-w [2] (2022)	2 410	**Segmentation**	**Industrial**	**Weak**	2	**Before-After**	No
WaRP-C [28] (2024)	10 406	Class./**Segm.**	**Industrial**	**Weak**	28	Materials	No
WS^2 (Ours)	**11 060**	**Segmentation**	**Industrial**	**Weak**	2	**Before-After**	**Yes**

Table 1. Summary comparison of WS² against existing waste-image datasets. For each dataset we report the number of images, the primary task (classification, detection or segmentation), the acquisition environment, the supervision regime, the number of labels and their annotation type, and whether video data are provided. Shared attributes between WS² and others are highlighted in **bold**, for a direct comparison of our contributions. WS² is the only dataset that combines *Before-After* weak supervision with video sequences for segmentation in an industrial setting, and with 11 060 frames offers more than four times the images of the only other industrial *Before-After* supervised collection (ZeroWaste-w, 2 410). Moreover, the table underscores the community's increasing emphasis on realistic, scalable benchmarks tailored to industrial environments and weakly supervised paradigms, owing to their enhanced broader applicability compared to alternative approaches.

WS conditions in recent times, and the only other datasets comparable to ours are WaRP[28] and Zerowaste[2].

While the WaRP [28] dataset comprises images acquired in a waste sorting plant and WS segmentation pipeline, it does not allow training models using *Before-After* supervision. The weak supervision in WARP is obtained by cropping the images of the belt to single objects of different materials and training a classifier to generate SMs directly for each specific material. ZeroWaste [2] is the first to exploit the HO's implicit *Before-After* supervision dynamic, but it contains a limited WS solutions benchmark and does not fully exploit the potential of WS methods given the limited set of 1200 images for each class. Our dataset offers significantly more images, with around 9600 images (4800 collected before and 4800 after the HO) for the training set and other 1500 images for the test set, from a modern plant addressing a more specific sorting task. Moreover, our images are collected as video frames, to promote the design of WS solutions leveraging temporal coherence like POF-CAM [14], while images in [2] are collected only sparsely.

3. Learning with Before-After Supervision

In this section, we detail the end-to-end pipeline we designed to learn with the *Before-After* supervision framework. Our goal is to obtain a semantic segmentation model that, given an image, either acquired *before* or *after* the HO, produces a binary mask accurately segmenting the *unwanted* items in the image. Given the highly occluded nature of the scene, we designed our pipeline to meet two key requirements: the resulting masks must exhibit both high semantic precision, ensuring that only truly *unwanted* items are identified, and fine-grained spatial accuracy, with masks boundaries closely aligning with the actual contours of the objects. The pipeline is composed of two stages, each addressing a specific task.

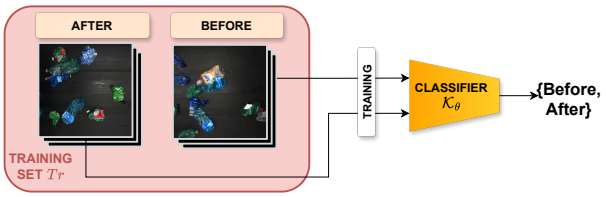


Figure 3. Overview of the first stage in our pipeline. We train the auxiliary classifier $\mathcal{K}_\theta$ on the training set $Tr = \{Tr^B, Tr^A\}$, to distinguish between *before* (Tr^B) and *after* (Tr^A) images. During the training, $\mathcal{K}_\theta$ learns to identify discriminative features associated with *unwanted* objects, since these are present only in the *before* images. Once trained, $\mathcal{K}_\theta$ is used to compute the *before* saliency maps $Sm_{bef}(Tr^B)$ on the original *before* training set, effectively highlighting the unwanted objects in the images.

Auxiliary Classifier Training: In the first stage (Figure 3), we leverage the *Before-After* separation of images to train an auxiliary classifier $\mathcal{K}_\theta$ to extract the visual features related to the unwanted objects. Let Tr^B and Tr^A denote the *before* and *after* subsets of our training set Tr, respectively. We train $\mathcal{K}_\theta$ on Tr, to distinguish between images in Tr^B and Tr^A. In doing so, $\mathcal{K}_\theta$ inherently learns to identify discriminative features associated with *unwanted* objects, since such objects appear only in Tr^B and represent a strong visual cue to solve the auxiliary task.

Pseudo-Mask Computation: In the second stage (Figure 4), we compute segmentation masks for the *un-*

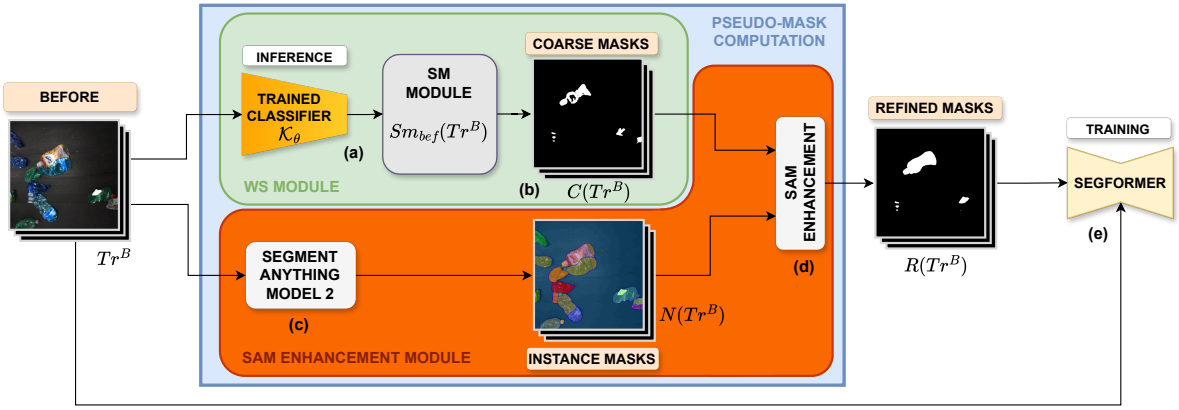


Figure 4. (a) In the WS module, we leverage the trained classifier $\mathcal{K}_\theta$ to compute the saliency maps of the *before* class $Sm_{\text{bef}}(Tr^B)$ over the *before* training set Tr^B. (b) We then threshold $Sm_{\text{bef}}(Tr^B)$ to produce initial *coarse masks* $C(Tr^B)$. (c) In parallel, in the SAM Enhancement Module, we employ SAM2 to generate *instance masks* $N(Tr^B)$. (d) We then refine $C(Tr^B)$ using the instance masks $N(Tr^B)$, following the procedure described in [4], resulting in *refined masks* $R(Tr^B)$. (e) Finally, we use $R(Tr^B)$ as pseudo-mask annotations to train a FS segmentation model on Tr^B, SegFormer [25].

wanted objects by leveraging the classifier $\mathcal{K}_\theta$ trained in the first stage. Specifically, as illustrated in the WS Module in Figure 4.a, we employ $\mathcal{K}_\theta$ to generate SMs for the *before* class, namely $Sm_{\text{bef}}(Tr^B)$, on each image in the *before* subset Tr^B. The maps $Sm_{\text{bef}}(Tr^B)$ highlight the most discriminative regions for the *before* class, which effectively correspond to the *unwanted* objects. By thresholding $Sm_{\text{bef}}(Tr^B)$, we obtain coarse binary masks $C(Tr^B)$ that provide initial estimates of the regions containing *unwanted* items (Figure 4.b). To improve the quality of these masks, we introduce a two-step SAM Enhancement Module. First, in Figure 4.c we apply SAM2 [19] to each image in Tr^B, producing fine-grained *instance masks* $N(Tr^B)$, where each individual object is segmented separately. Then, in Figure 4.d, we refine the the coarse masks $C(Tr^B)$ using the instance masks $N(Tr^B)$ by selecting SAM segments that overlap with the saliency regions, filling holes and refining boundaries following the procedure described in [4]. This results in high-fidelity *refined masks* $R(Tr^B)$. Finally, we use the refined masks $R(Tr^B)$ as pseudo-mask annotations to train a FS segmentation model on the *before* images Tr^B (Figure 4.e).

4. Background Removal Three-Classes Training Strategy

Let Λ be the set of classes $\Lambda = \{before, after\}$ and let $Tr^\lambda \in \{Tr^B, Tr^A\}$ be the subset of training images labeled as $\lambda \in \Lambda$. In our framework, *before* and *after* frames are acquired by cameras under slightly different light conditions. As a result, images I_i^λ of the same class set Tr^λ share nearly identical background appearances, whereas backgrounds differ significantly between *before* and *after* image sets. As shown in Figure 6.a, this particular light condition introduces a strong bias for the auxiliary classifier $\mathcal{K}_\theta$ which, when trained directly on the full training set Tr under the standard *Before-After* procedure (Section 3), tends to identify the background cues as distinctive features rather than the objects differences. In fact, the SM of the *before* class shown in Figure 6 highlights the background reflex as relevant for the class. A classifier trained in this way becomes completely useless in any WS segmentation solution, as it fails to localize the actual *unwanted* objects. To mitigate this problem, we implemented a novel three-class training strategy based on Background Removal (BR).

To unbind the background information from class identity, we compute for each image I_i^λ a binary foreground/background mask $M^{bg}(I_i^\lambda)$ using robust statistical estimation. We then use these masks to construct a new training set $Br(Tr)$ for the auxiliary classifier $\mathcal{K}_\theta$, consisting of three categories: (i) background-masked *before* images, (ii) background-masked *after* images, and (iii) background-only images, where foreground objects have been masked out. In this way, we expand the dataset to a third class of images consisting solely of background images, shifting from the binary set Λ to the three-class set $\hat{\Lambda} = \{before, after, background\}$. The resulting $Br(Tr)$, shown in Figure 5.b, is then used to train $\mathcal{K}_\theta$ instead of the original Tr. As illustrated in Figure 6.b, modeling the background as an explicit class reduces the likelihood that $\mathcal{K}_\theta$ will rely on background features when generating saliency maps for the *before* class. To maintain class balance during training, background-only images are computed from half of the *before* images and half of the *after* images.

To compute background binary masks $M^{bg}(I_i^\lambda)$ we rely on two assumptions: (i) all images within the same class share an almost identical background, so foreground re-

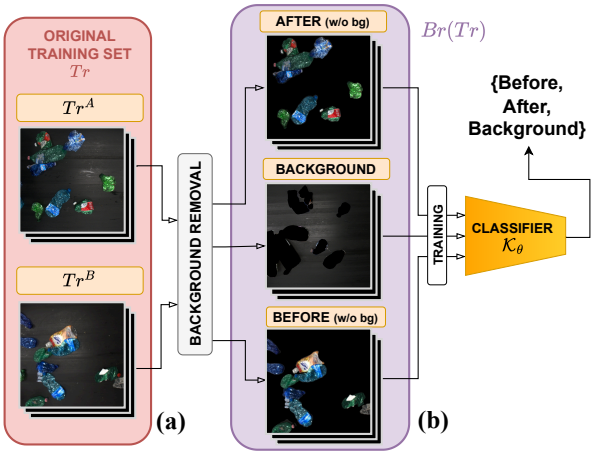


Figure 5. The original training set Tr, consisting in *before* (Tr^B) and *after* (Tr^A) images, undergo the BR process described in Section 4. As output, we get a new background-removed training set $Br(Tr)$ of three classes: one containing only the extracted background images and the other two representing the *before* and *after* images without background. These three classes (*before*, *after*, and *background*) are used to train the auxiliary classifier $\mathcal{K}_\theta$.

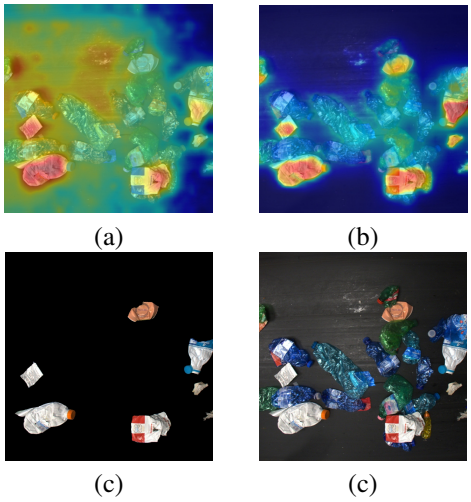


Figure 6. (a) *Before* image saliency map ($Sm(I^B)$) from a classifier (POF-CAM [14]) trained with standard *before-after* images with the background included (as in Figure 3). (b) Saliency map from the same classifier trained with the BR three-classes-strategy (as in Figure 5). (c) Ground Truth mask. (d) Original image. Training $\mathcal{K}_\theta$ on the three-class $Br(Tr)$ leads to more accurate SM to localize the *unwanted* items than training it on Tr.

gions can be identified as areas that significantly deviate from the pixel-wise median image of that class; (ii) since the background (the conveyor belt) is predominantly gray with white reflexes, pixels with high color saturation are more likely to belong to foreground objects. The detailed description of the BR procedure to compute the binary masks is detailed in Section 1 of the supplementary material.

5. The WS^2 Dataset

In this section, we introduce the WS^2 Dataset, specifically designed for training WS segmentation models for sorting process. The dataset captures a real-world conveyor belt where only colored semi-transparent PET objects are selected from a stream of mixed plastics, providing high-resolution video acquired both *before* and *after* the HO's activity. Section 5.1 details the acquisition campaign and setup, while Section 5.2 describes the dataset's structure.

5.1. Data Collection

Our dataset was acquired in the context described in Section 1. Two cameras (Blackfly S BFS-PGE-200S6C), having resolution 5472 x 3648 pixels, were positioned to capture images *before* and *after* the HO (as in Figure 1). The cameras were fixed on ceiling-mounted supports one meter above the conveyor belt and approximately two meters apart from each other, to ensure stability and avoid vibrations of the belt or interference with the work of the HO.

Given the high resolution and the large field of view of the cameras, it was necessary to crop images to capture only relevant sections of the conveyor belt. Images were cropped to 1000 x 1000 pixels to cover the entire belt width while excluding the HOs and their work area, which might introduce a bias when training $\mathcal{K}_\theta$. The belt moved at approximately 1 m/s, and videos were acquired at 12 fps to maintain a temporal relation between consecutive frames. Additionally, we reduced the exposure time and increased the gain to avoid motion blur and acquire sharp and bright images. Images were saved in JPEG format to optimize storage without significantly compromising their quality.

5.2. The WS^2 and the Waste Sorting Task

We selected a conveyor belt processing PET materials, including various types of PET: transparent, bluish, and opaque. The HOs leave only semi-transparent colored PET *wanted* objects on the belt, removing all the other *unwanted* objects. Consequently, the *before* images capture the initial mixed material flow, while the *after* images contain exclusively selected semi-transparent colored PET objects. The dataset comprises 11060 images, divided into 313 video sequences for the *after* class and 284 for the *before* class. Comprehensively, training and validation sets comprise 4712 *after* and 4851 *before* unlabeled images.

The test set comprises 1001 *after* and 496 *before* images, all annotated at the pixel level solely for performance assessment. Expert annotators trained in recognizing *wanted* and *unwanted* objects manually drew a bounding box around *unwanted* objects, and let SAM (Segment

Anything Model) [19] produce segmentation maps. SAM supports diverse input prompts like points, boxes, or masks, making it adaptable to various tasks, including WS segmentation. Segmentation masks were then manually refined at a pixel level by annotators, resulting in a labeled test set for semantic segmentation. We deliberately limited manual annotations to the test set for two reasons. First, the annotation process is extremely labor-intensive and costly, making full-dataset labeling prohibitive. Second, by constraining annotations to the evaluation split, we underscore the viability of WS learning as a cost-effective alternative for training automatic recognition models.

Finally, while the *after* images primarily contain semi-transparent colored PET objects, occasional anomalies may appear, as HO can sporadically overlook some items. However, although our dataset's labels reflect human performance, such errors are random and infrequent rather than systematic. Therefore, our annotations remain sufficiently reliable as these anomalies can be treated as casual noise when training $\mathcal{K}_\theta$. Using robust deep learning models can correct these occasional mistakes and exceed human selection performance. With this dataset, we aim to foster further research into WS methods in industrial sorting.

6. Experiments

In our experimental setup we benchmark WS^2 by comparing the segmentation performance of diverse SM methods based on Class Activation Maps (CAMs), focusing on the segmentation of *unwanted* objects. CAMs are interpretability tools that, in the context of image classification, highlight the contribution of each pixel to a specific class [29]. All methods are trained using the BR three-class strategy described in Section 4, even if testing is performed on images with the background included. This avoids the risk of discarding useful information in test images during BR. Using the same setting, all experiments with standard two-class training consistently underperformed (qualitative comparison in Figure 6). We divded the selected tested methods in three distinct groups.

The first group includes traditional CAM methods built upon a standard classifier, such as GradCAM,[21], GradCAM++ [3] and LayerCAM [9]. These methods were selected as they do not require any special training procedure and simply operate on the auxiliary classifier $\mathcal{K}_\theta$. The main drawback is that a pixel p belonging to an object of class λ, might not receive a high classification score when it does not belong to a relevant classification pattern, resulting in small segmentation masks. We trained the classifier $\mathcal{K}_\theta$ used for these methods using a categorical cross-entropy loss and an Adam optimizer at a learning rate of 5x10^{-4}.

The second group includes methods that incorporate additional learning constraints, such as spatial and temporal consistency, by introducing additional reconstruction losses during the classifier training to enhance the SMs [10, 14]. Specifically, PuzzleCAM [10] integrates a *puzzle module* that divides the image into non-overlapping patches and enforces a reconstruction loss between the CAM of the full image and the combined CAMs of the patches, encouraging the global SM to retain fine-grained local details. Puzzle Optical Flow CAM (POF-CAM) [14] is a WS video-segmentation method that extends [10] by incorporating temporal information from video sequences, using optical flow between consecutive frames within the *before* and *after* streams to enhance the CAMs. These methods are trained according to their default setups: SGD optimizer and a multi-label soft margin classification loss. We tuned the hyperparameters for each method and, for the training losses described in [10, 14], we set $\alpha = 2$ and $\beta = 6$.

The third category includes the transformer-based CAM method WeakTr [30], which first computes coarse CAMs from patch token embeddings using a convolutional layer, then refines them using self-attention maps. Both coarse and refined CAMs are jointly used for classification, enabling image-level supervision to yield detailed SMs. We replace WeakTr's online refinement with the SAM-based refinement for a fair comparison with other methods.

We assess segmentation performances using the mean Intersection over Union (IoU), computed across various processing stages, on the annotated test set. For each method, we used a fixed 80/20 train/validation split and a maximum of 25 training epochs with early stopping to prevent overfitting. Images are resized to 512×512 and processed with a batch size of 32. We apply random color jitter augmentation to slightly modify brightness, contrast and saturation. We employed ResNet50 as backbone for all methods, except for WeakTr, which requires a transformer-based backbone, for which we used a DeIT [23].

To ensure consistency and reliability, all experiments focus on segmenting *unwanted* objects in the same test set Ts, which is never used for training or validation at any stage. We assess the accuracy of the segmentation masks returned at different levels of our pipeline for each method on both *before* and *after* images of Ts, namely Ts^B and Ts^A, to highlight the performance difference for each class. The three distinct levels of our benchmark of the state-of-the-art are defined as follows:

- **(i) Coarse Masks C(Ts) mIoU**: We measure the mIoU directly on the coarse masks $C(Ts)$ derived by the *before* CAM $M_{bef}(Tst)$ generated by the classifier $\mathcal{K}_\theta$ on both Tst^A, $Tst^B \subset Tst$.
- **(ii) Refined Masks R(Ts) mIoU**: We refine the coarse masks $C(Tst)$ using the Sam Enhancement Module illustrated in Figure 4.b. Instance masks $N(Tst)$ are computed by SAM2 [19] and used by the algorithm described in [4] to obtain the refined masks $R(Tst)$. The mIoU for this stage is measured on $R(Tst)$.

(iii) SegFormer Masks S(Ts) mIoU: We measure the mIoU of the segmentation masks $S(Tst)$ returned from SegFormer [25] trained on Tr^B and $R(Tr^B)$.

Table 2 reports the mIoU scores at each evaluation stage, while Figure 7 shows qualitative examples of saliency maps produced by different WS methods before thresholding into $C(Ts)$, alongside the corresponding original images and ground-truth masks.

Models	$C(Ts)$		$R(Ts)$		$S(Ts)$	
	Ts^B	Ts^A	Ts^B	Ts^A	Ts^B	Ts^A
GradCAM [21]	19.86	8.95	25.03	12.67	27.64	14.10
GradCAM++ [3]	16.40	6.54	20.08	9.94	25.74	7.72
LayerCAM [9]	17.69	9.3	21.31	12.70	28.10	14.10
PuzzleCAM [10]	33.52	15.29	35.35	16.99	37.15	13.74
POF-CAM [14]	**38.70**	**20.93**	**41.40**	**23.08**	**42.58**	**19.78**
WeakTr [30]	21.44	8.20	23.73	9.62	24.14	4.42

Table 2. Mean Intersection Over Union (mIoU) percentage (%) of SOTA on *before* and *after* Images and in three stages: raw CAM, after SAM-Refinement and after SegFormer. POF-CAM[14], using temporal consistency, significantly outperforms other methods.

Since WS methods identify the regions of an image I that are most relevant for $\mathcal{K}_\theta$ to assign I to a specific class, and since the *unwanted* object represents the key region for the *before* class, segmentation masks in *after* images leads to poor performance. This happens because SMs attempt to identify *before* features in images that belong to a different class. This performance gap between the two classes persists even in the masks estimated in the next refinement steps. Furthermore, the resulting class imbalance and the limited accuracy of pseudo-masks can induce negative learning in the SegFormer network, which can't effectively disentangle the concept of *unwanted* objects from the *before* class, leading to inconsistent generalization. As a result, while segmentation performance steadily improves on *before* images, it may degrade on *after* ones.

Another remarkable piece of evidence is the impact of the SAM refinement, improving the performance of each method by several mIoU points. This shows that the level of detail brought by vision foundation models remains difficult to attain by WS segmentation alone methods alone. Nonetheless, the choice of the WS model remains crucial. Table 2 shows that the ranking of the best-performing models on our task remains very similar after refinement, consolidating the fact that, even with refinement, highly detailed SMs are needed to obtain good masks using SAM, as some methods perform better than others at localizing semantically relevant regions. This becomes especially true in use cases like waste sorting, where we can appreciate a considerable distribution shift with respect to the datasets used to train foundation models used for refinement.

Remarkably, we can observe how the methods employing additional learning constraints [10, 14] outperform the others. In particular, the spatial consistency provided by the puzzle module helps to detect multiple objects of the same class scattered across the image, which is the standard setting in our dataset. POF-CAM [14], the only method that leverages temporal consistency, significantly outperforms all other solutions, highlighting the critical role of temporal information in a dataset of this nature, a factor absent in previous works. Notably, WeakTr, despite being state-of-the-art in CAM generation, performs poorly on our dataset compared to methods employing puzzle modules. This highlights its limited generalization capability for specific tasks relative to convolutional-based approaches.

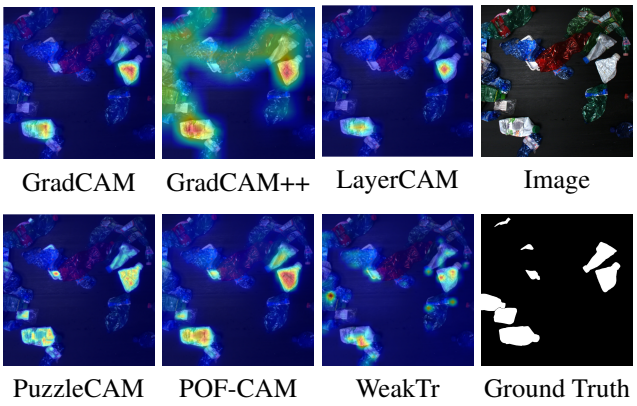


Figure 7. Qualitative examples of SMs obtained using different WS methods on an image taken from the test set compared with the original RGB image and the ground truth mask.

7. Conclusions

We introduced WS^2, the first large-scale, multiview video dataset for weakly supervised segmentation in industrial waste sorting. Our dataset captures over 11 000 "before" and "after" video frames around a manual removal operation on a plastic processing conveyor belt. To take advantage of the *Before-After* supervision implicit in the human removal operation, we developed a three-stage pipeline in which we (i) train an auxiliary classifier to localize *unwanted* items via saliency maps, (ii) refine these maps using SAM, and (iii) use the resulting refined masks to train a fully supervised segmentation network, whithout the need of any manual annotation. To mitigate the background bias intrinsic in this problem we also implemented an innovative background removal-based three-class training strategy. Our extensive benchmarks, spanning saliency maps methods of different categories, demonstrate the critical roles of temporal continuity and background bias mitigation in improving weakly supervised segmentation accuracy for this task. Our work not only eases the development of better saliency map solutions but also opens avenues for integrating temporal coherence in video-based weak supervision and skilled activity understanding across diverse domains.

References

[1] Dustin Aganian, Benedict Stephan, Markus Eisenbach, Corinna Stretz, and Horst-Michael Gross. Attach dataset: Annotated two-handed assembly actions for human action understanding. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11367–11373. IEEE, 2023. 3

[2] Dina Bashkirova, Mohamed Abdelfattah, Ziliang Zhu, James Akl, Fadi Alladkani, Ping Hu, Vitaly Ablavsky, Berk Calli, Sarah Adel Bargal, and Kate Saenko. Zerowaste dataset: Towards deformable object segmentation in cluttered scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21147–21157, 2022. 2, 3, 4

[3] Aditya Chattopadhay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *2018 IEEE winter conference on applications of computer vision (WACV)*, pages 839–847. IEEE, 2018. 7, 8

[4] Tianle Chen, Zheda Mai, Ruiwen Li, and Wei-lun Chao. Segment anything model (sam) enhanced pseudo labels for weakly supervised semantic segmentation. *arXiv preprint arXiv:2305.05803*, 2023. 5, 7

[5] Debanjan Datta, Nathan Self, John Simeone, Amelia Meadows, Willow Outhwaite, Linda Walker, Niklas Elmqvist, and Naren Ramakrishnan. Timbersleuth: Visual anomaly detection with human feedback for mitigating the illegal timber trade. *Information Visualization*, 22(3):223–245, 2023. 3

[6] Michael S Fulton, Jungseok Hong, and Junaed Sattar. Trashicra19: A bounding box labeled dataset of underwater trash. 2020. 3

[7] Jungseok Hong, Michael Fulton, and Junaed Sattar. Trashcan: A semantically-segmented dataset towards visual detection of marine debris. *arXiv preprint arXiv:2007.08097*, 2020. 3, 4

[8] Baoxiong Jia, Ting Lei, Song-Chun Zhu, and Siyuan Huang. Egotaskqa: Understanding human tasks in egocentric videos. *Advances in Neural Information Processing Systems*, 35: 3343–3360, 2022. 3

[9] Peng-Tao Jiang, Chang-Bin Zhang, Qibin Hou, Ming-Ming Cheng, and Yunchao Wei. Layercam: Exploring hierarchical class activation maps for localization. *IEEE Transactions on Image Processing*, 30:5875–5888, 2021. 7, 8

[10] Sanghyun Jo and In-Jae Yu. Puzzle-cam: Improved localization via matching partial and full features. In *2021 IEEE international conference on image processing (ICIP)*, pages 639–643. IEEE, 2021. 7, 8

[11] Maria Koskinopoulou, Fredy Raptopoulos, George Papadopoulos, Nikitas Mavrakis, and Michail Maniadakis. Robotic waste sorting technology: Toward a vision-based categorization system for the industrial robotic separation of recyclable waste. *IEEE Robotics & Automation Magazine*, 28(2):50–60, 2021. 3, 4

[12] Nikhil Venkat Kumsetty, Amith Bhat Nekkare, Sowmya Kamath, et al. Trashbox: trash detection and classification using quantum transfer learning. In *2022 31st Conference of Open Innovations Association (FRUCT)*, pages 125–130. IEEE, 2022. 3, 4

[13] Xudong Lin, Fabio Petroni, Gedas Bertasius, Marcus Rohrbach, Shih-Fu Chang, and Lorenzo Torresani. Learning to recognize procedural activities with distant supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13853–13863, 2022. 3

[14] Andrea Marelli, Luca Magri, Federica Arrigoni, and Giacomo Boracchi. Temporal-consistent cams for weakly supervised video segmentation in waste sorting. In *European Conference on Computer Vision*. Springer, 2025. 2, 4, 6, 7, 8

[15] Kosuke Moriwaki, Gaku Nakano, and Tetsuo Inoshita. The brio-ta dataset: Understanding anomalous assembly process in manufacturing. In *2022 IEEE International Conference on Image Processing (ICIP)*, pages 1991–1995. IEEE, 2022. 3

[16] Harsh Panwar, PK Gupta, Mohammad Khubeb Siddiqui, Ruben Morales-Menendez, Prakhar Bhardwaj, Sudhansh Sharma, and Iqbal H Sarker. Aquavision: Automating the detection of waste in water bodies using deep transfer learning. *Case Studies in Chemical and Environmental Engineering*, 2:100026, 2020. 3, 4

[17] PF Proença and P Simões. Taco: Trash annotations in context for litter detection. arxiv 2020. *arXiv preprint arXiv:2003.06975*, 2003. 3, 4

[18] Francesco Ragusa, Antonino Furnari, Salvatore Livatino, and Giovanni Maria Farinella. The meccano dataset: Understanding human-object interactions from egocentric videos in an industrial-like domain. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1569–1578, 2021. 3

[19] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 5, 7

[20] Tim J Schoonbeek, Tim Houben, Hans Onvlee, Fons Van der Sommen, et al. Industreal: A dataset for procedure step recognition handling execution errors in egocentric videos in an industrial-like setting. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 4365–4374, 2024. 3

[21] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017. 7, 8

[22] Joao Sousa, Ana Rebelo, and Jaime S Cardoso. Automation of waste sorting with deep learning. In *2019 XV Workshop de visao Computacional (WVC)*, pages 43–48. IEEE, 2019. 3, 4

[23] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pages 10347–10357. PMLR, 2021. 7

[24] Maria Triassi, Rossella Alfano, Maddalena Illario, Antonio Nardone, Oreste Caporale, and Paolo Montuori. Environmental pollution from illegal waste disposal and health effects: A review on the "triangle of death". *International Journal of Environmental Research and Public Health*, 12 (2):1216–1236, 2015. 1

[25] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in neural information processing systems*, 34: 12077–12090, 2021. 5, 8

[26] Zihui Xue, Yale Song, Kristen Grauman, and Lorenzo Torresani. Egocentric video task translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2310–2320, 2023. 3

[27] Mindy Yang and Gary Thung. Classification of trash for recyclability status. *CS229 project report*, 2016(1):3, 2016. 3, 4

[28] Dmitry Yudin, Nikita Zakharenko, Artem Smetanin, Roman Filonov, Margarita Kichik, Vladislav Kuznetsov, Dmitry Larichev, Evgeny Gudov, Semen Budennyy, and Aleksandr Panov. Hierarchical waste detection with weakly supervised segmentation in images from recycling plants. *Engineering Applications of Artificial Intelligence*, 128:107542, 2024. 3, 4

[29] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016. 7

[30] Lianghui Zhu, Yingyue Li, Jiemin Fang, Yan Liu, Hao Xin, Wenyu Liu, and Xinggang Wang. Weaktr: Exploring plain vision transformer for weakly-supervised semantic segmentation. *arXiv preprint arXiv:2304.01184*, 2023. 7, 8

[31] Dimitri Zhukov, Jean-Baptiste Alayrac, Ramazan Gokberk Cinbis, David Fouhey, Ivan Laptev, and Josef Sivic. Cross-task weakly supervised learning from instructional videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3537–3545, 2019. 3