StreetView-Waste: A Multi-Task Dataset for Urban Waste Management
Supplementary Material

Diogo J. Paulo[1,2], João Martins[1], Hugo Proença[1,2], João C. Neves[1,3]

[1]University of Beira Interior, Portugal [2]IT: Instituto de Telecomunicações [3]NOVA LINCS

diogo.paulo@ubi.pt

Abstract

This supplementary material is organized as follows:

- *Section 1 details an extended analysis of current state-of-the-art datasets;*
- *Section 2 shows more illustrative examples of StreetView-Waste for the three main tasks: waste container detection, waste container tracking, and waste overflow segmentation;*
- *Section 3 provides additional dataset statistics, including per-class spatial distributions of container centers and camera intrinsics/distortion parameters;*
- *Section 4 provides additional results, in this case, ablation studies for our geometry-aware strategy on the waste overflow segmentation task;*
- *Section 5 shows more experimental results regarding the waste container tracking and counting task;*
- *Section 6 analyzes model complexity for the waste segmentation task, including speed–accuracy trade-offs, GFLOPs, VRAM, and latency.*

1. State-Of-The-Art Datasets Analysis

To contextualize our contribution, Table 1 provides a comprehensive comparison between StreetView-Waste and other prominent public datasets for waste analysis. Our dataset is unique in its combination of large scale (over 36,000 images), a focus on the challenging street-level fisheye perspective common in municipal vehicles, and multi-task annotations that explicitly support detection, instance segmentation, and, crucially, multi-object tracking. While many datasets exist for waste classification or detection in controlled or underwater environments, StreetView-Waste is the first to provide a dedicated benchmark for the operational challenges of urban waste logistics.

2. Dataset Examples

To highlight the diversity and difficulty of our dataset, Figure 2 presents a selection of challenging examples for each of our three main tasks. These images go beyond the introductory examples in the main paper to highlight the specific difficulties that models must overcome in this real-world vehicular context.

The provided examples highlight the need for models that are robust to a wide range of lighting conditions and scales. The presence of frequent, partial occlusions from other vehicles is a constant challenge that requires models to recognize objects from incomplete visual information. Furthermore, the sequence demonstrates that this is not a simple linear motion problem. StreetView-Waste has scenarios with a significant appearance-change problem, which can confuse trackers that rely heavily on visual similarity. This validates our choice to benchmark with a motion-first tracker like ByteTrack [28]. Regarding waste overflow segmentation, these images show why this is a particularly difficult task. Unlike typical instance segmentation tasks with well-defined objects (e.g., cars, people), overflowing litter is often amorphous, lacks clear boundaries, and can blend seamlessly with complex urban textures like asphalt, grass, and concrete. This motivates our exploration of geometric priors to provide additional cues for separating foreground litter from the background plane.

3. Dataset Additional Statistics

Figure 1 depicts the per-class spatial distribution of container centers in order to help identify potential (unintended) biases and better understand the challenges of our dataset. Across classes, centers concentrate in a vertical band on the roadside half of the image, with a slight upward bias in the y-axis (consistent with flank-mounted cameras and curved fisheye projection). Classes with smaller physical size (e.g., *Battery Container*) show wider dispersion and higher vertical placement. In contrast, large street bins (e.g., *Default/Blue/Green/Yellow*) occupy a tighter band with larger areas in the y-axis than in the x-axis. *Biodegradable* exhibits the narrowest spread, indicating fewer viewpoints or more constrained placement. These patterns confirm that StreetView-Waste captures realistic operational

Table 1. **Comparative analysis of public waste datasets.** Our work, StreetView-Waste, fills a critical gap by providing a large-scale, multi-task benchmark specifically for the urban street-level context from a vehicular perspective.

Name	Classes	#Images	Task	Context
TrashCan 1.0 [13]	4	7,212	Detection, Segmentation	Underwater
Trash-ICRA19 [8]	3	5,700	Detection	Underwater
TACO [20]	60	1,500	Segmentation	Waste in the Wild
UAVVaste [16]	1	772	Segmentation	Aerial-View of Abandoned Waste
Trashnet [27]	6	2,527	Classification	Controlled Scenario
WaDaBa [5]	3	4,000	Detection	Controlled Scenario of Plastic Waste
GLASSENSE-VISION [24]	7	2,000	Classification	Indoor
Waste Classification Data v2 [21]	3	27,500	Classification	Kaggle/Mixed Waste
Waste Images from Sushi Restaurant [2]	16	500	Classification	Indoor/Restaurant
Open litter map [18]	11	>100,000	Classification	Crowdsourced/GPS-tagged
Drinking Waste Classification	4	9,640	Detection	Controlled-Scenario of Drinking Waste
spotgarbage [19]	1	2,400	Classification	Public spaces
DeepSeaWaste [10]	5	3,055	Classification	Underwater Waste
MJU-Waste v1.0 [25]	1	2,475	Segmentation	Indoor
Domestic Trash Dataset	10	>9,000	Classification	Waste in the Wild
Cigarette butt dataset [14]	1	2,200	Detection, Segmentation	Cigarette butt in the Wild
TrashBox [17]	7	17,785	Detection	Controlled Scenario
PortlandStateSingh RECYCLE [22]	5	11,500	Classification	Controlled Scenario
TIDY [9]	9	304	Classification	Waste in the Wild
Garbage Dataset (V2)	10	19,762	Classification	Web Controlled-Scenario
RealWaste [23]	9	4,752	Classification	Public spaces
BePLi Dataset v1 [12]	1	3,709	Detection, Segmentation	Controlled-Scenario
Zerowaste [4]	4	4,503	Detection	Industrial-Grade Waste
LOTS [3]	1	3,572	Segmentation	Waste on the Sand
StreetView-Waste (Ours)	**8**	**36,478**	**Detection, Segmentation, Tracking**	**Urban Street Level**

geometry rather than a uniform image-plane prior, highlighting its scientific value.

Camera intrinsics and distortion (per flank camera). Here, we present in Table 2 the camera intrinsic and distortion parameters. *Note:* $\mathbf{K}_{\alpha=0}$ and $\mathbf{K}_{\alpha=1}$ correspond to OpenCV fisheye undistortion with $\alpha = 0$ (tighter crop) and $\alpha = 1$ (full FOV).

Table 2. **Camera intrinsics and fisheye distortion parameters.**

Intrinsics (f_x, f_y, c_x, c_y)				
Matrix	f_x	f_y	c_x	c_y
$\mathbf{K}$	683.74	683.53	1007.45	529.18
$\mathbf{K}_{\alpha=0}$	719.71	719.49	959.50	539.50
$\mathbf{K}_{\alpha=1}$	352.28	352.17	959.50	539.50
Fisheye distortion (k_1, k_2, k_3, k_4)				
	k_1	k_2	k_3	k_4
Coefficients	-0.03631	-0.00908	$+0.00734$	-0.00272

4. Ablation Studies for Waste Segmentation Task

We conduct ablation studies focusing on one of the most challenging tasks, the waste overflow segmentation, to investigate the impact of surface normals and depth maps on the performance of our method. These results provide insight into how different model architectures utilize this information. For clarity, we present the results alongside the baseline performance (RGB only) and the full method (RGB + Depth + Normals) in Table 3.

From Table 3, we draw two key conclusions. On the one hand, surface normals are the more valuable cue: For nearly every model, adding only surface normals yields a

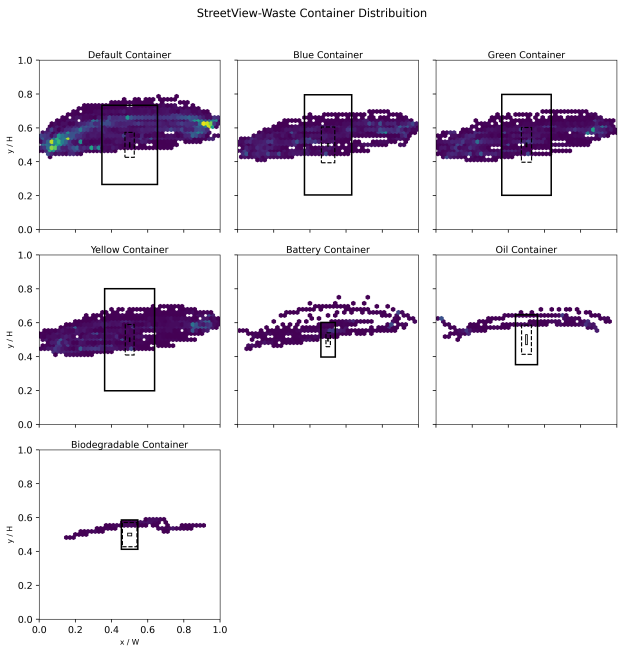

Figure 1. **Per-class spatial distribution of container centers.** Each panel shows the normalized center (x_n, y_n) for all instances of a class in StreetView-Waste. The rectangles are the interquartile range (IQR) along each axis. The concentration in a curb-side vertical band and the class-dependent spread (e.g., tighter for large bins, wider for small containers) reflect the flank-mounted fisheye setup and real curb-side placement.

Table 3. **Ablation study for geometric priors on the waste overflow segmentation task (mAP@0.5).** This table isolates the impact of adding only depth versus only surface normals. Surface normals provide a stronger performance boost than depth alone.

Model	RGB	RGB+Depth	RGB+Normals	RGB+Depth+Normals
YOLOv11 [15]	0.50±0.01	0.53±0.01	**0.54±0.01**	0.52±0.01
YOLACT [6]	0.41±0.02	0.50±0.02	**0.56±0.01**	0.52±0.02
Mask2Former [7]	0.18±0.02	0.14±0.02	0.23±0.03	**0.29±0.02**
Mask R-CNN [11]	**0.41±0.02**	0.12±0.01	0.28±0.01	0.12±0.01
SOLOv2 [26]	**0.20±0.03**	0.07±0.01	0.15±0.02	0.07±0.01

greater performance benefit than adding only depth. This suggests that the local orientation information provided by normals is more effective at distinguishing amorphous litter from its surroundings than the raw distance information from the depth map. On the other hand, depth can be detrimental, for complex architectures like Mask R-CNN [11] and SOLOv2 [26], adding depth information results in a performance drop compared to both the RGB-only baseline and the normals-only version. This reinforces the hypothesis from the main paper that the estimated depth map may introduce noise that disrupts the pre-trained features of these models, whereas transformer-based models like Mask2Former [7] are better equipped to attend to useful information across different modalities selectively.

5. Waste Container Tracking and Counting

To further analyze the impact of our tracking heuristics, Tables 4 and 5 present a per-class breakdown of counting performance. The results show that the heuristics provide significant improvements for common, frequently occluded classes, such as green and blue containers. Notably, for the default container using ByteTrack [28], the complete set of heuristics (H_1+H_2+H_3) performs better than the intermediate step (H_1+H_2), demonstrating that the spatial constraint (H_3) is crucial for preventing incorrect merges of visually similar containers. This nuanced view confirms that a combination of temporal and spatial constraints is helpful for robust counting across diverse scenarios.

Table 4. **Per-class counting performance with added heuristics using ByteTrack [28].** Applying heuristics incrementally leads to notable improvements across most container classes.

Class	Heuristic	MAE↓	SAD↓	RMSE↓	MAPE↓
Yellow Container	Baseline	0.43	9	0.95	88.89%
	H_1	0.19	4	0.53	33.33%
	$H_1 + H_2$	0.05	1	0.22	11.11%
	$H_1 + H_2 + H_3$	0.05	1	0.22	11.11%
Green Container	Baseline	1.05	22	3.35	244.44%
	H_1	0.14	3	0.38	33.33%
	$H_1 + H_2$	0.10	2	0.31	22.22%
	$H_1 + H_2 + H_3$	0.14	3	0.38	33.33%
Default Container	Baseline	0.48	10	0.93	30.56%
	H_1	0.24	5	0.65	16.67%
	$H_1 + H_2$	0.33	7	0.65	22.22%
	$H_1 + H_2 + H_3$	0.24	5	0.58	17.36%
Blue Container	Baseline	1.14	24	3.19	250.00%
	H_1	0.24	5	0.58	44.44%
	$H_1 + H_2$	0.10	2	0.31	16.67%
	$H_1 + H_2 + H_3$	0.10	2	0.31	22.22%
Oil Container	Baseline	0.05	1	0.22	33.33%
	H_1	0.05	1	0.22	33.33%
	$H_1 + H_2$	0.05	1	0.22	33.33%
	$H_1 + H_2 + H_3$	0.05	1	0.22	33.33%
Battery Container	Baseline	0.19	4	0.44	66.67%
	H_1	0.14	3	0.38	50.00%
	$H_1 + H_2$	0.14	3	0.38	50.00%
	$H_1 + H_2 + H_3$	0.14	3	0.38	50.00%
Biodegradable Container	Baseline	0.14	3	0.49	100.00%
	H_1	0.05	1	0.22	100.00%
	$H_1 + H_2$	0.00	0	0.00	0.00%
	$H_1 + H_2 + H_3$	0.00	0	0.00	0.00%
Overall	Baseline	3.48	73	7.80	82.96%
	H_1	1.05	22	1.93	24.96%
	$H_1 + H_2$	0.76	16	1.75	17.77%
	$H_1 + H_2 + H_3$	**0.71**	**15**	**1.70**	**16.03%**

6. Model Complexity for the Waste Segmentation Task

Figure 3 summarizes the speed–accuracy trade-off, Table 6 reports parameter count, theoretical compute (GFLOPs), and peak inference memory, and Figure 4 shows per-model latency. The values are measured and rounded to two decimals; in some cases, differences across channels are below

Figure 2. **Additional examples from the StreetView-Waste dataset.** (First Column) Waste container detection examples showing variations in lighting, container type, and partial occlusion by street elements. (Middle Column) A sequence for waste container tracking, illustrating how a container's appearance changes due to the vehicle's motion. (Last Column) Examples for waste overflow segmentation, highlighting the challenge of delineating amorphous litter from complex backgrounds.

Table 5. **Per-class counting performance with added heuristics using BoT-SORT [1].** Applying heuristics incrementally leads to notable improvements across most container classes.

Class	Heuristic	MAE↓	SAD↓	RMSE↓	MAPE↓
Yellow Container	Baseline	1.43	30	2.58	288.89%
	H_1	0.29	6	0.69	50.00%
	$H_1 + H_2$	0.14	3	0.49	22.22%
	$H_1 + H_2 + H_3$	0.14	3	0.49	22.22%
Green Container	Baseline	1.62	34	3.07	366.67%
	H_1	0.14	3	0.38	22.22%
	$H_1 + H_2$	0.10	2	0.31	11.11%
	$H_1 + H_2 + H_3$	0.14	3	0.38	22.22%
Default Container	Baseline	1.62	34	2.79	136.81%
	H_1	0.29	6	0.69	19.44%
	$H_1 + H_2$	0.24	5	0.58	17.36%
	$H_1 + H_2 + H_3$	0.24	5	0.58	17.36%
Blue Container	Baseline	1.10	23	2.44	233.33%
	H_1	0.19	4	0.44	38.89%
	$H_1 + H_2$	0.05	1	0.22	11.11%
	$H_1 + H_2 + H_3$	0.10	2	0.31	22.22%
Oil Container	Baseline	0.05	1	0.22	33.33%
	H_1	0.05	1	0.22	33.33%
	$H_1 + H_2$	0.05	1	0.22	33.33%
	$H_1 + H_2 + H_3$	0.05	1	0.22	33.33%
Battery Container	Baseline	0.38	8	0.87	133.33%
	H_1	0.14	3	0.38	50.00%
	$H_1 + H_2$	0.14	3	0.38	50.00%
	$H_1 + H_2 + H_3$	0.14	3	0.38	50.00%
Biodegradable Container	Baseline	0.19	4	0.69	100.00%
	H_1	0.10	2	0.31	100.00%
	$H_1 + H_2$	0.10	2	0.31	100.00%
	$H_1 + H_2 + H_3$	0.10	2	0.31	100.00%
Overall	Baseline	6.43	135	9.60	187.61%
	H_1	1.19	25	1.91	27.19%
	$H_1 + H_2$	**0.81**	**17**	**1.36**	**16.03%**
	$H_1 + H_2 + H_3$	0.90	19	1.79	17.96%

0.1, making them appear constant in the table. From Figure 3, the most accurate setting is YOLACT [6] with seven channels, followed by YOLOv11seg [15]. Thus, making them the highest-accuracy/highest-throughput models when compared to models with more complex architectures.

Regarding the cost of adding channels, Table 6 shows that GFLOPs rise only slightly with channel count. Latency overhead, however, depends on the architecture (Figure 4): it is modest for YOLOv11seg [15] and YOLACT [6], but substantial for SOLOv2 [26] and two-stage/transformer models.

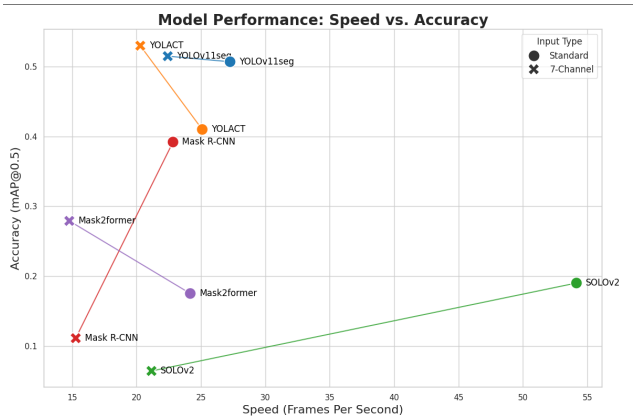

Figure 3. **Model performance comparison (Speed vs. Accuracy).** It is observed the trade-off between inference speed (frames per second) and accuracy (mAP@0.5) across different segmentation models (YOLOv11seg [15], YOLACT [6], SOLOv2 [26], Mask R-CNN [11], and Mask2Former [7]) under both standard and 7-channel input conditions.

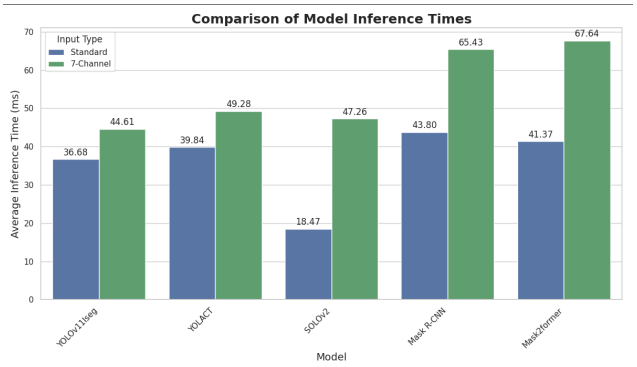

Figure 4. **Model inference time comparison.** The average inference times (in milliseconds) of the same models under standard and 7-channel input types are compared, highlighting the computational overhead introduced by additional channels.

Table 6. **Model complexity across input channels.** Computed at 640×640, batch size $= 1$. VRAM is peak usage during inference.

Model Complexity @ 640×640 (Batch=1)				
Model	**Channels**	**Params (M)**	**GFLOPs**	**VRAM (MB)**
YOLOv11 [15]	3	27.59	141.90	275.81
	4	27.59	142.00	278.37
	6	27.59	142.20	282.50
	7	27.59	142.40	283.07
Mask R-CNN [11]	3	43.92	207.27	1730.56
	4	43.92	220.93	3102.72†
	6	43.93	207.64	1515.52
	7	43.93	209.48	1802.56
SOLOv2 [26]	3	46.23	222.00	395.00
	4	46.23	222.00	342.00
	6	46.24	222.00	447.00
	7	46.24	222.00	447.00
YOLACT [6]	3	30.60	148.25	212.97
	4	30.60	148.89	214.55
	6	30.61	150.18	217.70
	7	30.61	150.82	219.27
Mask2Former [7]	3	47.37	119.99	1423.00
	4	47.37	120.04	1424.00
	6	47.37	120.13	1425.00
	7	47.37	120.18	1426.00

†4-channel Mask R-CNN shows an unusually high peak VRAM (3.03 GB = 3102.72 MB).

References

[1] Nir Aharon, Roy Orfaig, and Ben-Zion Bobrovsky. Botsort: Robust associations multi-pedestrian tracking. *arXiv preprint arXiv:2206.14651*, 2022. 5

[2] arthurcen. Waste images from sushi restaurant. https://www.kaggle.com/datasets/arthurcen/waste-images-from-sushi-restaurant, 2020. Kaggle dataset. 2

[3] Paola Barra, Alessia Auriemma Citarella, Giosué Orefice, Modesto Castrillón-Santana, and Angelo Ciaramella. Lots: Litter on the sand dataset for litter segmentation. In *18th International Conference on Machine Vision and Applications (MVA)*, pages 1–4. IEEE, 2023. 2

[4] 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

[5] Janusz Bobulski and Jacek Piatkowski. Pet waste classification method and plastic waste database-wadaba. In *International Conference on Image Processing and Communications*, pages 57–64. Springer, 2017. 2

[6] Daniel Bolya, Chong Zhou, Fanyi Xiao, and Yong Jae Lee. Yolact: Real-time instance segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9157–9166, 2019. 3, 5, 6

[7] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1290–1299, 2022. 3, 5, 6

[8] Michael Fulton, Jungseok Hong, Md Jahidul Islam, and Junaed Sattar. Robotic detection of marine litter using deep visual detection models. In *International Conference on Robotics and Automation (ICRA)*, pages 5752–5758. IEEE, 2019. 2

[9] gale31. Tidy: Trash image dataset "yucky". https://github.com/gale31/TIDY, 2023. Accessed on July 14, 2025. Dataset of classified trash images. 2

[10] Henry Haefliger. Deepseawaste. https://www.kaggle.com/datasets/henryhaefliger/deepseawaste, 2020. Kaggle dataset of approximately 3,055 underwater waste images with CSV annotations. 2

[11] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2961–2969, 2017. 3, 5, 6

[12] Mitsuko Hidaka, Koshiro Murakami, Kenta Koshidawa, Shintaro Kawahara, Daisuke Sugiyama, Shin'ichiro Kako, and Daisuke Matsuoka. Bepli dataset v1: Beach plastic litter dataset version 1 for instance segmentation of beach plastic litter. *Data in Brief*, 48:109176, 2023. 2

[13] Jungseok Hong, Michael Fulton, and Junaed Sattar. Trashcan: A semantically-segmented dataset towards visual detection of marine debris. *arXiv preprint arXiv:2007.08097*, 2020. 2

[14] Adam Kelly. Cigarette butt dataset. https://www.immersivelimit.com/datasets/cigarette-butts, 2018. 2,200 synthetic COCO-formatted images of cigarette butts on ground; accessed July 14, 2025. 2

[15] Rahima Khanam and Muhammad Hussain. Yolov11: An overview of the key architectural enhancements. *arXiv preprint arXiv:2410.17725*, 2024. 3, 5, 6

[16] Marek Kraft, Mateusz Piechocki, Bartosz Ptak, and Krzysztof Walas. Autonomous, onboard vision-based trash and litter detection in low altitude aerial images collected by an unmanned aerial vehicle. *Remote Sensing*, 13(5):965, 2021. 2

[17] Nikhil Venkat Kumsetty, Amith Bhat Nekkare, Sowmya Kamath, et al. Trashbox: trash detection and classification using quantum transfer learning. In *31st Conference of Open Innovations Association (FRUCT)*, pages 125–130. IEEE, 2022. 2

[18] Seán Lynch. OpenLitterMap.com – Open Data on Plastic Pollution with Blockchain Rewards (Littercoin). *Open Geospatial Data Software and Standards*, 3(1), 2018. 2

[19] Gaurav Mittal, Kaushal B Yagnik, Mohit Garg, and Narayanan C Krishnan. Spotgarbage: smartphone app to detect garbage using deep learning. In *Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 940–945, 2016. 2

[20] Pedro F Proença and Pedro Simões. Taco: Trash annotations in context for litter detection. *arXiv preprint arXiv:2003.06975*, 2020. 2

[21] sapal6. Waste classification data v2. https://www.kaggle.com/datasets/sapal6/waste-

classification-data-v2, 2020. Kaggle dataset. Extended version with non-recyclable ("N") class added. 2

[22] Suresh Singh, Minwei Luo, and Yu Li. Multi-class anomaly detection. In *International Conference on Neural Information Processing*, pages 359–371. Springer, 2022. 2

[23] Sam Single, Saeid Iranmanesh, and Raad Raad. Realwaste: a novel real-life data set for landfill waste classification using deep learning. *Information*, 14(12):633, 2023. 2

[24] Joan Sosa-García and Francesca Odone. "hands on" visual recognition for visually impaired users. *ACM Transactions on Accessible Computing (TACCESS)*, 10(3):1–30, 2017. 2

[25] Tao Wang, Yuanzheng Cai, Lingyu Liang, and Dongyi Ye. A multi-level approach to waste object segmentation. *Sensors*, 20(14):3816, 2020. 2

[26] Xinlong Wang, Rufeng Zhang, Tao Kong, Lei Li, and Chunhua Shen. Solov2: Dynamic and fast instance segmentation. *Advances in Neural Information Processing Systems*, 33:17721–17732, 2020. 3, 5, 6

[27] Mindy Yang and Gary Thung. Classification of trash for recyclability status. *CS229 project report*, 2016(1):3, 2016. 2

[28] Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Fucheng Weng, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. Bytetrack: Multi-object tracking by associating every detection box. In *European Conference on Computer Vision*, pages 1–21. Springer, 2022. 1, 3