One-shot logo detection for large video datasets and live camera surveillance in criminal investigations

Stefanos Demertzis [1], Sabina B. van Rooij [2], Michalis Lazaridis [1], Henri Bouma [2*], Manuel Álvarez Fernández [3], Johan-Martijn ten Hove [2], Rodrigo Sainz Méndez [3], Petros Daras [1]

[1] CERTH-ITI, Thessaloniki, Greece
[2] TNO, The Hague, The Netherlands
[3] Policia National, Madrid, Spain

ABSTRACT

Logos on clothing are sometimes one of the crucial clues to find a suspect in surveillance video. Automatic logo detection is important during investigations to perform the search as quickly as possible. This can be done immediately after an incident on live camera streams or retrospectively on large video datasets from criminal investigations for forensic purposes. It is common to train an object detector with many examples on a logo dataset to perform logo detection. To obtain good performance, the logo dataset must be large. However, it is time-consuming and difficult to obtain a large training set with realistic annotated images. In this paper, we propose a novel approach for logo detection that requires only one logo image (or a few images) to train a deep neural network. The approach consists of two main steps: data generation and logo detection. In the first step, the logo image is artificially blended in a person re-identification dataset to generate an anonymized synthetic dataset with logos on clothing. Various augmentation steps appeared to be necessary to reach a good performance. In the second step, an object detector is trained on the synthetic dataset, subsequently providing detections on recorded images, video files, and live streams. The results consist of a quantitative assessment based on an ablation study of the augmentation steps and a qualitative assessment from end users that tested the tool.

Keywords: Object detection, Deep learning, Surveillance, Video processing, Logo detection, One-shot learning.

1. INTRODUCTION

In criminal investigations, it is common to find a suspect in a surveillance video based on a short description of the clothing. Logos appear to be distinctive clothing characteristics with large effect on identification [Dysart, 2006]. Automatic logo detection can speed up the process for an investigator to retrieve the suspect in the video. The search can be performed immediately after the incident on live camera streams. Alternatively, it can be done retrospectively on large video datasets from criminal investigations for forensic purposes.

It is common to train an object detector with many examples [Jocher, 2021][Lin, 2017]. This detector can be fine-tuned on a logo dataset to perform logo detection [Rooij, 2022]. In order to reach good performance, a large dataset is required. However, it is time-consuming and difficult to obtain a large training set with realistic annotated images.

In this paper, we propose a new approach for logo detection. The main novelty is that it requires only one logo image (or a few images) to train a deep neural network. The contribution is the minimization of the effort and simplification of the process of creating logo detectors on demand, even on new, never shown before logos. The approach consists of two main steps: data generation and logo detection. In the first step, the logo image is artificially blended in a person re-identification dataset to generate an anonymized synthetic dataset with logos on clothing. Face anonymization is applied to minimize the processing of personal data [Rooijen, 2020]. Various augmentation steps appeared to be necessary to reach a good

[*] Henri.Bouma@tno.nl; phone +31 6 52 77 90 20

S. Demertzis, S.B. van Rooij, M. Lazaridis, H. Bouma, M. Álvarez Fernández, J.-M. ten Hove, R. Sainz Méndez, P. Daras, “One-shot logo detection for large video datasets and live camera surveillance in criminal investigations”, Proc. SPIE Artificial Intelligence for Security and Defence Applications, vol. 12742, (2023).
https://doi.org/10.1117/12.2681903

performance related to logo size, location, color, opacity, contrast, brightness, rotation, color jitter, and perspective transformation. In the second step, a Faster R-CNN detector with additional augmentations (cropping, scaling, and color jitter) and a YOLOv8 detector with the built-in augmentations are trained on the synthetic dataset. The whole framework, consisting of both the data generation and the logo detection is integrated into a single tool, able to train the system on demand for detecting logos in images, videos, and live streams, using minimal input.

The outline of this paper is as follows: Section 2 describes the method, Section 3 presents the experiments and results, and finally, Section 4, summarizes the conclusions.

2. METHOD

This section contains four subsections. First, we introduce an overview of the overall approach by presenting the system architecture (Sec. 2.1). Then we focus on the two main components: data generation (Sec. 2.2) and logo detection (Sec. 2.3). The last part describes the graphical user interface (Sec. 2.4).

2.1 System architecture

The system architecture is shown in Figure 1. To start the process, a user provides one (or a few) logo images to the system. The data generation component (Sec. 2.2) combines these logo images with a dataset containing images of persons to generate a new anonymized dataset with synthetic images. This synthetic dataset is then used to train the logo detector (Sec. 2.3), resulting in a trained deep neural network (DNN) model. This model can be applied to images and video data in order to detect the logos, referred to as ‘inference’. Finally, the filtered results are presented to the user through the graphical user interface (GUI, Sec. 2.4). Docker containers are used to facilitate portability of the system, as a stand-alone executable package of software that includes everything needed to run the application. The communication interfaces between the modules use HTTP POST and JSON messages. The system is implemented using the Python programming language and the PyTorch machine learning framework.

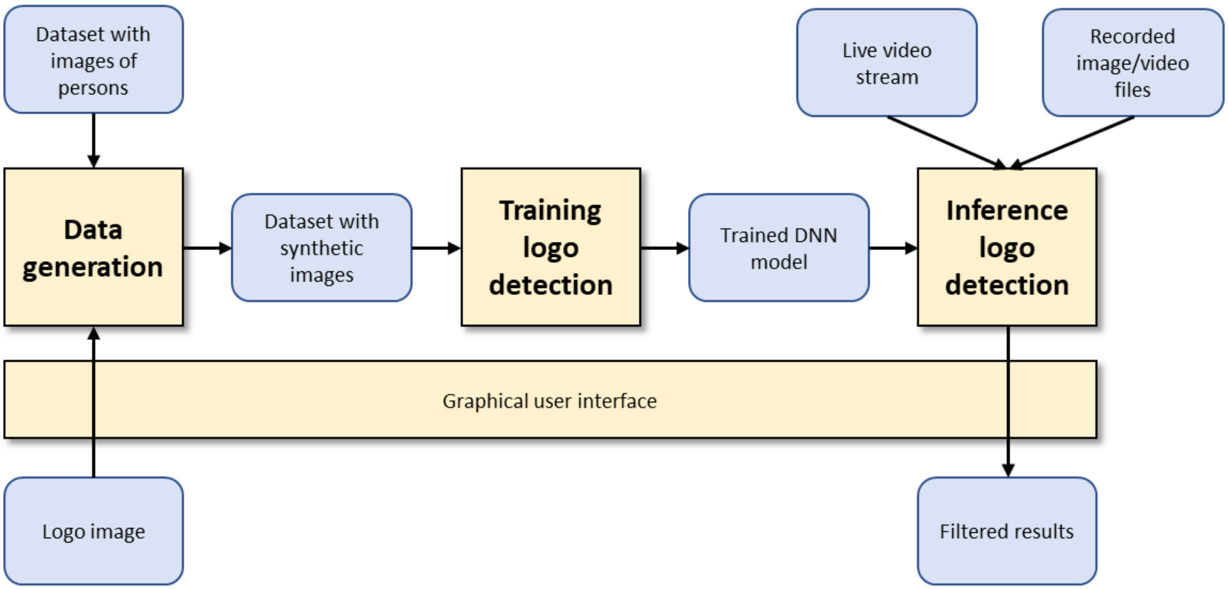


Figure 1: System architecture.

2.2 Dataset generation

As the input dataset with images of persons, we used MARS [Zheng, 2016], which is a large benchmark dataset for person re-identification. This is an extension of the commonly used Market-1501 dataset [Zheng, 2015]. MARS is collected with 6 cameras and it contains 1261 pedestrians that are recorded in at least 2 cameras and 3248 distractors. The size of the cropped persons in MARS is 256x128 pixels, which is more suitable than the size in Market-1501 (128x64 pixels). Further image resizing towards 512x256 pixels did not appear to further improve the quality of the system. Face anonymization is applied with a median filter to minimize personal data [Rooijen, 2020] (Figure 2a).

The system requires at least one logo to generate a dataset. When multiple logos are provided, they are selected randomly during the data generation process. Multiple logos are especially valuable when there are multiple versions (e.g., an old and a new version) of the same brand logo. The size of the output dataset is fixed and does not depend on the number of input logos. In order to perform blending, it is necessary that the image contains information in the alpha channel to create a transparent background and an opaque foreground. The blending is made more natural by making the logo foreground slightly transparent (with approximately 60% opacity).

Person segmentation is used to determine the location and size of the logo. Person segmentation is performed with FCN_Resnet50 [Long, 2015] and generates a binary mask (Figure 2b). A distance transform is computed on this mask to determine the center of the person and the maximum size of the logo. Randomness is added in the location and size to create a variety of logos. The logos are randomly resized between 0.4 times and 0.8 times the maximum size computed in the distance transform. A cropping augmentation was tried by randomly removing a part of the logo from the top, bottom, left, or right side. The probability of a crop was set to 20% (so the probability of a left crop is 5%) and the size of the crop is 35% of the image. Due to a deterioration in performance, the cropping was excluded from the proposed system.

There are two types of logos, which are handled in different ways. Monochrome logos can be generated with different colors (Figure 2c+d). This coloring is done based on the background such that there is maximum contrast between the colors of the logo and the clothing. The determination of the color with maximum contrast is done using the Delta E (CIE2000) of two colors. With a probability of 90% the logo is black or white, and with a probability of 10% the color is one of the selected other colors (including red, blue, green, yellow, and pink). Multi-color logos are just treated as they are without this color change.

TorchVision is used for several (other) augmentations. A random perspective transformation is applied with a distortion scale of 0.4 and a probability of 0.5. A random rotation is applied between -30 and +30 degrees. Color jitter was added with brightness, saturation, and contrast factors of 0.5 and a hue factor of 0.0.

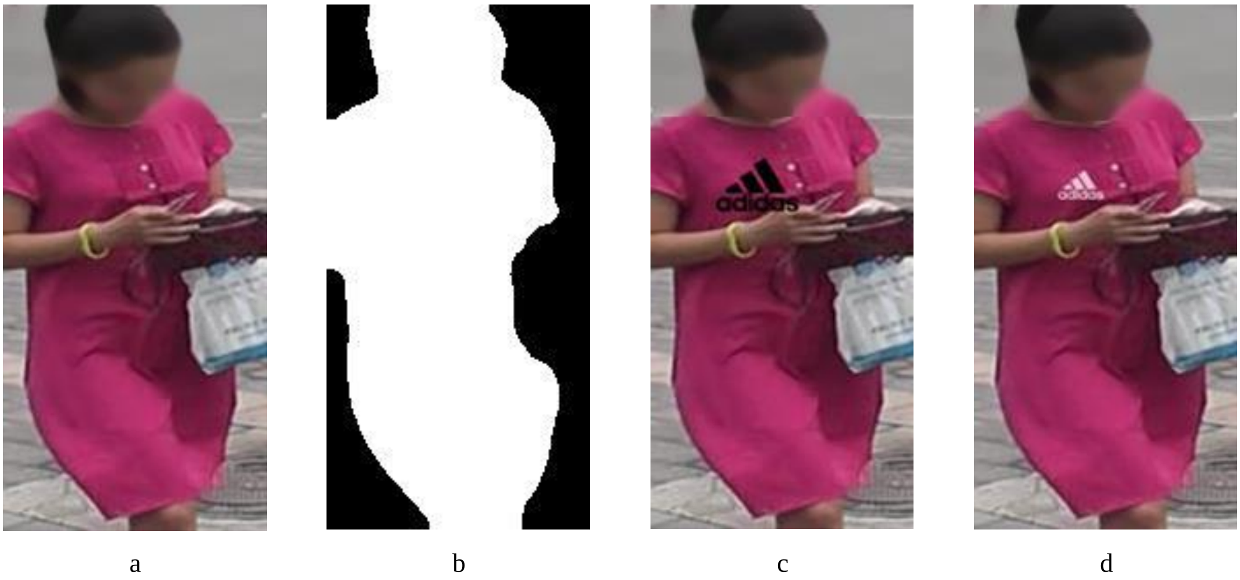


Figure 2: Synthetic image generation: (a) anonymized person without logo, (b) person segmentation, (c) logo artificially blended on the person, (d) logo with different color and size.

2.3 Logo detection

The system is capable of utilizing two object detectors: YOLOv8 and Faster R-CNN. YOLOv8 [Jocher, 2023] builds upon the success of its previous versions and introduces new features and improvements to enhance performance and flexibility. YOLOv8 is released under AGPL (GNU Affero General Public License) v3.0, so any derivative work must be distributed under the same or equivalent license terms. To create a generic modular solution, another object detector we experiment with is Faster R-CNN [Ren, 2015] with FPN (Feature Pyramid Network) [Lin, 2017]. Faster R-CNN is an object detection framework that employs a ResNet50 backbone architecture for feature extraction. The ResNet50 backbone is pre-trained on the ImageNet dataset, which enables the network to learn a rich set of features that can be used for a wide range of visual recognition tasks. Both networks exhibit comparable performance, with Faster R-CNN having a more permissive (MIT) license and YOLO demonstrating superior speed which is crucial for live-stream processing.

To improve the performance of the system, further data augmentation techniques have been used during training. While the built-in augmentations have been used with YOLOv8, additional ones have been implemented for Faster R-CNN, including Large Scale Jittering (LSE) [Ghiasi, 2021] and Color Jittering. LSE involves randomly scaling and cropping the input images to a target size of 512x512 pixels, with a scale range of 0.5 to 2, which helps to increase the diversity of the training data and prevent overfitting. This augmentation technique is especially helpful in detecting logos that appear at various scales. Color Jittering randomly changes the brightness, saturation, and hue of the images, which is especially helpful in improving the detection of logos under low-light conditions.

The network was trained on the synthetic dataset generated during the previous step, containing labeled images of people wearing different types of clothing with logos from various brands. For Faster R-CNN, the model was trained for 26 epochs using stochastic gradient descent (SGD) with a learning rate of 0.02. To prevent overfitting and allow the model to converge to a better solution, the learning rate was reduced by a factor of 10 on epochs 16 and 22. During inference, the input image is resized and padded to form a 512x512 square before being fed into the network. To address the real-time object detection challenge, an adaptive processing scheme has been implemented, particularly during the detection of logos in live streams. The detector processes as many frames as possible, keeping an eye on the real-time objective, making it possible to run the framework not only on high-end machines but also on more mainstream (GPU-equipped) systems.

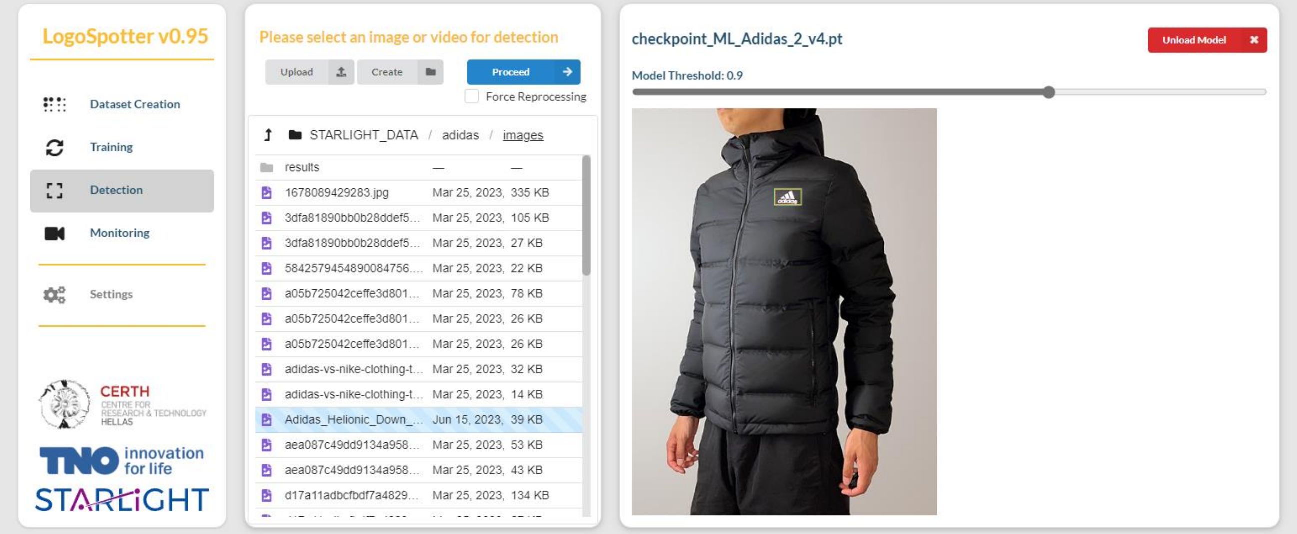
For YOLOv8-based implementation, the 25.9M-parameter YOLOv8m model was utilized. To optimize the training process, Stochastic Gradient Descent (SGD) was employed as the optimizer. After a 3-epoch learning rate warm-up strategy, the initial learning rate was set to 1e-3 and linearly decayed to 1e-5 over 200 epochs. Additionally, mosaic augmentation was incorporated, combining multiple images at different scales to enhance diversity. During inference, the input image is resized and padded to become a 640x640 square before being fed into the network.

Subsequently, the trained model is utilized by the system to detect and classify logos in images, videos, and live streams, achieving high accuracy.

2.4 Graphical user interface

The graphical user interface (GUI) is shown in Figure 3 and it supports the user to navigate through the workflow in the following steps, each with its dedicated pane within the GUI.

- Dataset creation: The dataset creation pane enables the user to upload one or multiple logo images, which are then utilized to generate a synthetic dataset.
- Training: In the training pane, the synthetic dataset is used to train the logo detector.
- Detection: The detection pane allows the user to select recorded image or video files and apply the logo detector to them. The GUI supports the user to find the relevant logos in the selected files. Especially with videos, the detected logos are tracked throughout the duration of the video.
- Monitoring: Within the monitoring pane, logo detection is applied to a live video stream. The detected scenes are presented on the interface as short video clips containing the tracked logos and the user is able to switch between the live stream and the results at will.



Please select an image or video for detection

□ Force Reprocessing

STARLIGHT_DATA / adidas / images

results	—	—
1678089429283.jpg	Mar 25, 2023,	335 KB
3dfa81890bb0b28ddef5...	Mar 25, 2023,	105 KB
3dfa81890bb0b28ddef5...	Mar 25, 2023,	27 KB
5842579454890084756....	Mar 25, 2023,	22 KB
a05b725042ceffe3d801...	Mar 25, 2023,	78 KB
a05b725042ceffe3d801...	Mar 25, 2023,	26 KB
a05b725042ceffe3d801...	Mar 25, 2023,	26 KB
adidas-vs-nike-clothing-t...	Mar 25, 2023,	32 KB
adidas-vs-nike-clothing-t...	Mar 25, 2023,	14 KB
Adidas_Helionic_Down_...	Jun 15, 2023,	39 KB
aea087c49dd9134a958...	Mar 25, 2023,	53 KB
aea087c49dd9134a958...	Mar 25, 2023,	43 KB
d17a11adbcfbdf7a4829...	Mar 25, 2023,	134 KB

checkpoint_ML_Adidas_2_v4.pt

Model Threshold: 0.9

Figure 3: Graphical user interface (Adidas image source by "Adenosine Triphosphate" on https://commons.wikimedia.org/wiki/File:Adidas_Helionic_Down_Jacket.jpg (CC BY-SA).

3. EXPERIMENTS AND RESULTS

This section contains three parts. First, we describe the experimental setup (Sec. 3.1). Then we present the quantitative results of the ablation study (Sec. 3.2). Finally, we show the qualitative evaluation of the end user (Sec. 3.3).

3.1 Experimental setup

For data generation, we used four monochrome Adidas logos (mountain, trefoil, mountain with Adidas text, and trefoil with Adidas text) and one multi-color Real-Madrid logo. These images were blended into the MARS dataset as described in Sec 2.2.

For evaluation, we used two independent datasets, each consisting of 100 images with an Adidas logo or 100 images with a Real-Madrid logo. These images were scraped from public sources, such as Pixabay or Wikimedia.

The performance metrics used to evaluate the system are the following:

- **mAP50:** The mean average precision (mAP) at an Intersection over Union (IoU) threshold of 50%. The average precision (AP) is the weighted sum of precisions at each threshold where the weight is the increase in recall. Precision measures how well true positives (TP) can be found in all positives (TP + FP).
- **Recall:** Measures how well true positives (TP) can be found in all that should have been positive (TP + FN).

The choice of mAP50 over other metrics, such as mAP50:95, commonly used in object-detection literature, was based on the needs of the end users, namely the Law Enforcement Agencies (LEAs). The LEAs are interested in identifying suspects wearing clothes with a specific logo on them. It is not necessary to achieve a precise overlap between the ground truth and the prediction bounding boxes. The recall is also a very important metric in this scenario since the LEAs prefer inspecting more candidates than necessary (i.e. more FPs) to avoid missing real occurrences of the suspect logo (i.e. more FNs).

The software is flexible and was tested on several computers with different CPUs (e.g., Intel Core i7, AMD Ryzen 7), GPUs (e.g., GTX TITAN X, RTX 2070, RTX 3090, at least 8 GB), and size of RAM (at least 16 GB).

The timing was tested on a computer equipped with an Intel i7 CPU and NVIDIA Geforce RTX 2070 Super GPU with 8 GB GPU memory and 16 GB RAM. The data generation of one MARS dataset with a synthetic logo took 5 minutes, training on this generated dataset lasted 4 hours, and the inference on one evaluation set of 100 images took approximately 2 seconds.

3.2 Ablation study

The proposed system is described in Section 2. In this experiment, we perform an ablation study by comparing the results of the proposed system with many variations, each having only one parameter difference.

The experimental results are presented in Table 1 and Table 2. Our proposed system (Exp. ID = 1) achieves optimal performance in terms of mAP$_{50}$ or Recall. Specifically for the monochrome case (Table 1), removing the perspective transformation (ID = 7) has no significant difference in mAP but results in a noticeable decline in recall. Adaptation of any of the other transformations deteriorates the mAP$_{50}$ value. Therefore, we can conclude that the system benefits from the YOLOv8 detector, color changes, transparency, localization on the clothing, perspective transformation, and rotations. The results for multi-color logo are shown in Table 2. The results are similar to the monochrome logo, although reduced opacity seems to have less effect. The proposed system seriously deteriorates without size variation (ID = 3) or random rotation (ID = 9). Ignoring the person segmentation (ID = 6) slightly increases the Recall value, however the mAP$_{50}$ value is significantly reduced.

Table 1: Results of the ablation study regarding the monochrome logos

ID	Experiment with monochrome logos	mAP$_{50}$ (%)	Recall (%)
1	Proposed system	88.2 ± 1.	**81.0 ± 3.**
2	Proposed system with FasterRCNN w FPN (instead of YOLOv8)	85.9	78.3
3	Proposed system trained with only medium size logo (without random resize)	80.2	70.9
4	Proposed system trained with only black logos (instead of color changes)	77.2	68.3
5	Proposed system trained only with 100% opacity (instead of 60% opacity)	87.3	77.4
6	Proposed system trained at random position in image (ignoring person segmentation)	81.6	70.0
7	Proposed system trained without perspective transformation (instead of with perspective transformation)	**88.8**	77,1
8	Proposed system trained with random crops (instead of without crops)	86.1	71.7
9	Proposed system trained with random rotations (instead of without random rotations)	82.6	73.3

Table 2: Results of the ablation study regarding the multi-color logos

	Experiment with multi-color logos	mAP$_{50}$ (%)	Recall (%)
1	Proposed system	**84.6 ± 1.**	75.3± 1.
2	Proposed system with FasterRCNN w FPN (instead of YOLOv8)	82.2	74.8
3	Proposed system trained with only medium size logo (without random resize)	70.7	61.1
4	N/A (color changes are not used for multi-color)	-	-
5	Proposed system trained only with 100% opacity (instead of 60% opacity)	84.6	74.3

6	Proposed system trained at random position in image (ignoring person segmentation)	81.6	**76.2**
7	Proposed system trained without perspective transformation (instead of with perspective transformation)	83.7	71.8
8	Proposed system trained with random crops (instead of without crops)	80.1	73.8
9	Proposed system trained with random rotations (instead of without random rotations)	72.8	70.8

3.3 Evaluation by end user

The evaluation of the system was performed by the National Police in Spain. The system was evaluated in a co-development (CODEV) cycle of the STARLIGHT project. The technical partners and end users collaborated intensively in a short period to design, develop, test, and improve the system. It started with a clear definition of the scope: detection of logos on clothing in the surveillance domain. The initial system was developed by technical partners over several months. It was tested by the end-user on an independent dataset and feedback was collected. The process continued in several iterations of developing, testing, collecting new feedback, and further improvements. Initial versions of the system showed suboptimal performance and missing functionalities. The initial suboptimal performance showed too many false positives (generating too many alerts) and false negatives (missing detections). Later versions showed improved performance by using the augmentation steps described in Section 2. The initial system worked on recorded images and recorded video, but functionality for real-time live processing was missing. Later versions included new functionalities, such as real-time video processing of live streams, tracking in video, and detection of full-color logos.

4. CONCLUSION AND FUTURE WORK

We showed a novel approach for logo detection that requires only one logo image (or a few logo images) to train a deep neural network. The approach consists of two main steps: data generation and logo detection. The results consist of a quantitative assessment from an ablation study of the augmentation steps and a qualitative assessment from end users that tested the tool. The ablation study showed that the proposed system benefits from the YOLOv8 detector, color changes, transparency, localization on the clothing, perspective transformation, and rotations. The system can be used to detect and classify logos in images, videos, and live streams, with high accuracy. The proposed system can be useful in criminal investigations, such as identifying suspects or tracking their movements using CCTV footage.

Future work could include the use of transformers for one-shot object detection (e.g., [Chen, 2021]).

ACKNOWLEDGEMENTS

The work described in this paper is performed in the H2020 project STARLIGHT ("Sustainable Autonomy and Resilience for LEAs using AI against High priority Threats"). This project has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 101021797. Dataset generation (Sec. 2.2) was developed by TNO and object detection (Sec. 2.3) and graphical user interface (Sec. 2.4) were developed by CERTH.

REFERENCES

[1] Akhgar, B., Bayerl, P., Bailey, K., et al., "Accountability principles for artificial intelligence (AP4AI) in the internal security domain," Europol/CENTRIC, (2022).

[2] Chen, D. J., Hsieh, H. Y., & Liu, T. L., "Adaptive image transformer for one-shot object detection," IEEE CVPR, 12247-12256 (2021).

[3] Dysart, J., Lindsay, R., Dupuis, P., "Show-ups: the critical issue of clothing bias," Appl. Cognit. Psychol. 20, 1009-1023 (2006).

[4] Ghiasi, G., et al., "Simple copy-paste is a strong data augmentation method for instance segmentation," IEEE CVPR, 2918-2928 (2021).

[5] Jocher, G., et al., "YOLOv5," github.com/ultralytics/yolov5, (2021).

[6] Jocher, G., et al, "YOLOv8," https://github.com/ultralytics/ultralytics, (2023).

[7] Lin, T.-Y., et al., "Feature pyramid networks for object detection," IEEE CVRP, 2117-2125 (2017).

[8] Long, J., Shelhamer, E., & Darrell, T., "Fully convolutional networks for semantic segmentation," IEEE CVPR, 3431-3440 (2015).

[9] Ren, S., et al., "Faster R-CNN: Towards real-time object detection with region proposal networks," Advances in neural information processing systems 28, (2015).

[10] Rooij, S. van, Bouma, H., Mil, J. van, Hove, J.M., "Federated tool for anonymization and annotation in image data," Proc. SPIE 12275, (2022).

[11] Rooijen, A. van, Bouma, H., Pruim, R., Baan, J., Uijens, W., Mil, J., van, "Anonymized person re-identification in surveillance cameras," Proc. SPIE 11542, (2020).

[12] Zheng, L., Bie, Z., Sun, Y., Wang, J., Su, C., Wang, S., & Tian, Q., "MARS: A Video Benchmark for Large-Scale Person Re-identification", ECCV, (2016).

[13] Zheng, L., Shen, L., Tian, L., Wang, S., Wang, J., & Tian, Q., "Scalable person re-identification: A benchmark," IEEE ICCV, 1116-1124 (2015).