Combination of Object Tracking and Object Detection for Animal Recognition

Francis Williams
School of CSEE
Bangor University
Bangor, UK
eeub05@bangor.ac.uk

Ludmila I. Kuncheva
School of CSEE
Bangor University
Bangor, UK
l.kuncheva@bangor.ac.uk

Juan J. Rodríguez
Departamento de Ingeniería Informática
Universidad de Burgos
Burgos, Spain
jjrodriguez@ubu.es

Samuel L. Hennessey
School of CSEE
Bangor University
Bangor, UK
sml18vly@bangor.ac.uk

***Abstract*—While methods for object detection and tracking are well-developed for the purposes of human and vehicle identification, animal identification and re-identification from images and video is lagging behind. There is no clarity as to which object detection methods will work well on animal data. Here we compare two state-of-the art methods which output bounding boxes: the MMDetector and the UniTrack video tracker. Both methods were chosen for their high ranking on benchmark data sets. Using a bespoke pre-annotated database of five videos, we calculated the Average Precision (AP) of the outputs from the two methods. We propose a combination method to fuse the outputs of MMDetection and UniTrack and demonstrate that the proposed method is capable of outperforming both. *Index Terms*—Animal identification, Bounding boxes , Object tracking, Object detection.**

I. Introduction

Animal detection and tracking is an important research area in the current era of global climate change and increasing number of endangered species [1]. A lot of research effort has been invested into tracking humans [2] [3] [4] and vehicles [5] [6] [7] in comparison to animal tracking and recognition due to high demand related to autonomous vehicles, crowd control and more [8].

Object detection in images and videos of animals is the first step towards recognising species and identifying individual animals. This can be done by applying an object detector to the available images, one image at a time, and storing the bounding boxes or the masks of the detected animals. If a video is available, bounding boxes can be extracted with multi-object tracking (MOT) [9]. In both cases, the result is a collection of bounding boxes with individual animals. Bespoke object detectors are fine-tuned to recognise a wide range of objects in different poses, based on the appearance of the objects in the image [10]. Conversely, tracking methods rely mostly on identifying consistent trajectories of the bounding boxes, and only partly on appearance [11] [12]. MOT methods, however, due to the continuity of the trajectories, may identify bounding boxes where single-image detectors may fail.

In this study we are interested in comparing the accuracy of the two detection routes by matching the obtained bounding boxes with a ground truth in a quest to propose a fusion method which outperforms both. To this end, we run an experiment with five annotated video clips of fish, pigeons and pigs.

The rest of the study is organised as follows. The detection, tracking and evaluation methods are explained in Section II. Our proposed combination method is explained in Section III. The data and the experiment are detailed in Section IV. Section V gives our conclusion.

II. Related Work

A. Object Detection (MMDet)

The object detector adopted in this study was sourced from the site "Papers with Code"[1], which hosts the most recent developments in Machine Learning, and often gives a running comparison between the results reported in the papers on benchmark datasets. Following the best practices, we chose the MMDetection object detector (MMDet) [13].

Figure 1 shows an illustration of the steps involved in two-stage bounding box detection. At Step 1, the input image goes into the 'backbone', in our case ResNet-50, which transforms the image into feature maps. At Step 2, the feature maps are passed into their respective layers in the 'neck'. We used Feature Pyramid Network (FPN), which refines and re-configures the raw feature maps. This step is the link between the 'backbone' and 'heads'. Step 3 is the DenseHead which operates on the dense parts from the feature maps. This feeds into the RoI Head. Some of the feature maps were passed into this head directly, bypassing the DenseHead. These and the feature maps altered in Step 3 are used by the 'head' to predict what is in the image based on the labels the detector has been trained on.

B. Tracking (UniTrack)

UniTrack [14] is one of the highest ranking methods on the benchmark datasets used by the MOT community. It is a

This work is supported by the UKRI Centre for Doctoral Training in Artificial Intelligence, Machine Learning and Advanced Computing (AIMLAC), funded by grant EP/S023992/1. This work is also supported by the Junta de Castilla León under project BU055P20 (JCyL/FEDER, UE), the Ministry of Science and Innovation under project PID2020-119894GB-I00 co-financed through European Union FEDER funds, and the Ministry of Universities under mobility grant PRX21/00638.

[1]https://paperswithcode.com/

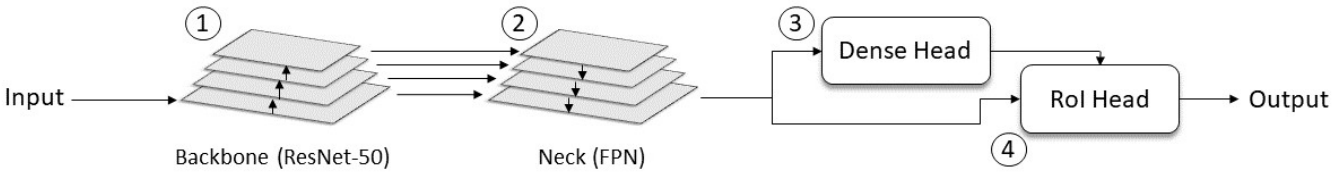


Fig. 1: Illustration of MMDet pipeline (reproduced from [13]).

diverse framework with multiple applications, including Single Object Tracking (SOT), Video Object Segmentation (VOS) and Multiple Object Tracking (MOT). For our experiments, we are interested in the MOT aspect of the framework.

There are two stages to this tracking process for MOT. Firstly, an Appearance Model is used to convert the 2-dimensional video frame into a feature map. In this instance, we used the 'default' recommended YOLOX detector. Secondly, Association is used to pair the output from the previous step with those in adjacent frames to create tracks. The tracker computes a distance matrix between existing tracks and new detections. The Hungarian algorithm is used to determine pair matches in adjacent frames. Again, the 'default' setting was used, in this case Imagenet-Resnet18-s3. The output of the MMDetector are the bounding boxes for each frame along with the corresponding track number as a label.

C. Evaluation

A unified view on the evaluation metrics for object detection has been recently proposed by Padilla et al. [15]. In a further study, Padilla et al. [16] observe that different object detection metrics have been applied in various competitions and benchmark studies, depending on the dataset and the detection method, giving and example of 14 such metrics. They subsequently provide an open-source performance metric repository available at https://github.com/rafaelpadilla/Object-Detection-Metrics#different-competitions-different-metrics.

In most applications, a detected bounding box is considered a match for a ground truth bounding box (true positive) if the intersection-over-union (IoU) value exceeds a given threshold. The recommended (and most widely-reported) metric of the quality of an object detector is the Average Precision (AP) at IoU = 0.5. It is calculated as the area under a precision-recall curve, constructed by varying a confidence threshold τ. This threshold is outputted by the detector for each bounding box. We use τ to either accept or reject a bounding box. First, the obtained thresholds in the whole collection of detected bounding boxes are sorted in descending order. The curve is obtained by scanning τ from largest to smallest. For each fixed τ, we consider only the bounding boxes with confidence greater than τ.

In our case, we assume that there is only one class (one type of animal in each video). If we were interested in multiple identities, AP would be calculated for each class separately, and the *mean* average precision mAP would be returned. In this study, we use the standard AP @IoU = 0.5.

III. The proposed fusion method

Our preliminary experiment showed that the detector returns duplicate bounding boxes. Occasionally, it also returns inadequately small bounding boxes. Therefore, we set up a percentile threshold P and removed the smallest $P\%$ bounding boxes from the detector output.

The next phase is aggregating the bounding boxes from the two outputs. To complete this phase, we apply the following steps for each frame t:

1) Identify the bounding boxes in frame t returned by the detector. Denote this list by B_{det}. Identify the bounding boxes in frame t returned by the tracker. Denote this list by B_{tr}. Pool together the two lists into a single list $B = B_{\text{det}} \cup B_{\text{tr}}$.
2) Calculate a square matrix M with IoU values between all pairs of bounding boxes in B. Set the main diagonal of M to zeros to eliminate the match between each box with itself.
3) Apply a duplicate threshold D on the values of M. All pairs of bounding boxes whose IoU is greater than D are perceived to be the same bounding box. This transforms M into a binary matrix M_b.
4) Considering M_b as an adjacency matrix of a graph, identify the connected components. Each component is fused into a single bounding box. The fusion takes the minimum top left corner (on both coordinates) and the maximum bottom right corner (on both coordinates). The detector output contains a value of certainty attached to each bounding box, while the tracker output places the same certainty to all boxes. To calculate the certainty of a fused bounding box (connected component), we take the maximum certainty of the boxes being fused.

The parameters of our combination methods are the percentile threshold P and the duplicate threshold D. Bellow we carry out grid-like experiments to demonstrate that the proposed method is capable of outperforming both the detector and the tracker taken individually.

IV. Experiment

A. Data

The data used in the experiment consist of five videos of animals: koi fish [2], pigeons (ground)[3], pigeons (kerb)[4], pigeons (square)[5] and pigs[6], available from www.pixabay.com. The video clips are free for commercial use according to the Pixabay license. Examples of images from the five videos are shown in Figure 2. These videos are an accurate representation of what one might obtain as video data from the field, e.g. loss of focus, occlusion, and camera movement.

To create ground truth, the videos were manually annotated by drawing a bounding box around each animal. This was done through www.makesense.ai. Links to the videos, their annotated versions, the annotations files and code are available at https://github.com/LucyKuncheva/Animal-Identification-from-Video. The characteristics of the data are described in Table I.

B. Purpose of the experiment.

The first part of the experiments is aimed at comparing MMDetector and UniTrack as bounding box detectors from animal videos.

The second part of the experiment compares the combination method with the two competitors.

C. Comparison between the Detector and the Tracker

MMDetection (MMDet) and the UniTrack were applied to the five videos. For fairness of test both methods were run using their default settings and configurations as outlined in the installation instructions.

a) Illustration of the mismatch: Figure 3 shows an example of a mismatch between the two approaches. We calculated the Average Precision for the two individual frames. Plots (a) and (b) show a frame where the detector (blue boxes) matches the ground truth (green boxes) a lot better than the tracker (red boxes). In contrast, in the frame in plots (c) and (d), the tracker matches the ground truth better. This illustrates the reason for our experiment, and the intuition behind combining the two approaches.

Counter-intuitively, $AP = 100$ for the detector in the top image, even though there are 6 bounding boxes, while the ground truth has three. Apparently, the AP metric on a single frame ignores near duplicates or the small bounding boxes which happen to be just noise here.

b) Counts per frame analysis: Table II shows details of the output of the tracker and the detector. We included the ground truth for comparison. It can be seen that the detector often overestimates the bounding box count.

Denote by $k(t)$ the number of ground truth bounding boxes in frame t, and by $k_{\text{det}}(t)$ and $k_{\text{tr}}(t)$, the counts for the detector and the tracker, respectively. To examine the match of the bounding box counts per frame further, we plot in Figure 4 k, k_{det}, and k_{tr} versus the frame number. The curves have been smoothed with a window of size 40 frames. It can be seen that the detector almost everywhere gives a larger value than the tracker, that is $k_{\text{det}}(t) > k_{\text{tr}}(t)$. It can also be observed the tracker values are typically closer to the ground truth, apart from those for video Pigeons (curb), where $k_{\text{det}}(t)$ yields a closer match.

As observed in part a), there may be noise or duplicates returned by the detector. This may account for the larger number of bounding boxes per frame. However, as in the illustration in Figure 3, the AP values may not be affected that much by the bb count.

c) Comparison of Average Precision (AP): AP was calculated for the detector and the tracker for the five videos. The results are displayed in Figure 5. Interestingly, even though the tracker returned more accurate number of bounding boxes in each frame, the average precision metric favoured the detector in all videos apart from Koi fish.

D. The combination method

The combination method was applied to the detector and the tracker outputs for the five videos. We decided to run a grid-like experiment in order to explore systematically different parameter combinations and demonstrate that the propose method is capable of achieving higher AP than each of the competitor outputs taken separately.

We used sets of values for $P = \{1, 3, 5, 10, 15, 20, 25, 30\}$ and for $D = \{0.40, 0.45, 0.50, ..., 0.95\}$. An example of the output of the combination method is shown in Figure 6.

Figure 6 demonstrates that the AP values obtained through the combination method are larger than the values for the detector and the tracker for most combinations of parameter values. Figure 7 shows the surfaces for the remaining four videos. While the improvement on the two competitors is not as pronounced as in the example in Figure 6, it is visible for the pigeon and the pigs videos. For the Koi fish video, the surface of the combination method peeks above the tracker AP at 0.5736 only for $P = 30\%$ and $D = 0.50$.

V. Conclusion

This work compares state-of-the-art object detector MMDetection (MMDet) and a multi-object tracker (UniTrack) as methods to extract salient bounding boxes from animal videos. As a comparison metric, we chose Average Precision (AP) @IoU = 0.5. We ran an experiment with five pre-annotated animal video clips sourced from Pixabay. Our results did not identify a clear winner. While the tracker produced more accurate number of bounding boxes per frame, the detector gave better average precision (AP) on four of the five videos. We subsequently propose a combination method which fuses near duplicates of bounding boxes from the two outputs and removes a given proportion of bounding boxes that are perceived to be inadequately small. Our results show that the combination method is capable of outperforming the two state-of-the-art methods.

[2] www.pixabay.com/videos/koi-carp-fishes-ornamental-fish-5652/

[3] www.pixabay.com/videos/pigeons-doves-and-pigeons-bird-city-4927/

[4] www.pixabay.com/videos/pigeons-eating-nature-birds-food-8234/

[5] www.pixabay.com/videos/birds-street-pigeon-29033/

[6] www.pixabay.com/videos/pigs-farm-animals-livestock-49651/

(a) Koi fish (b) Pigeons (ground) (c) Pigeons (curb) (d) Pigeons (square) (e) Pigs

Fig. 2: Examples of frames from the five videos.

TABLE I: Characteristics of the videos

Video	k	l	N	c	Min p/f	Max p/f	Avr p/f
Koi fish	536	22	1635	9	1	6	3.1
Pigeons (ground)	600	24	3079	17	3	8	5.1
Pigeons (curb)	443	17	4700	14	8	13	10.6
Pigeons (square)	300	9	4892	27	1	23	16.3
Pigs	500	16	6184	26	4	20	12.4

Table notes: k is the number of frames; l is the video length in seconds; N is the number of objects (individual animal clips); c is the number of classes (animal identities; not used here); Min p/f is the minimum number of animals per frame (image); Max p/f and Avr pf are respectively the maximum and the average numbers.

TABLE II: Details of the outputs of the detector (det) and the tracker (tr), The ground truth values (gt) are also included for comparison.

Video	Minimum per frame			Maximum per frame			Average per frame		
	gt	det	tr	gt	det	tr	gt	det	tr
Koi fish	1	1	0	6	11	6	3.1	5.3	2.1
Pigeons (ground)	3	2	1	8	15	7	5.1	6.8	4.6
Pigeons (curb)	8	3	1	13	16	11	10.6	8.8	6.6
Pigeons (square)	1	14	13	23	28	24	16.3	20.2	18.6
Pigs	4	8	2	20	37	18	12.4	20.4	9.6

It will be interesting to relate the parameter values to the characteristics of the video which may lead to insights about the best combination of values.

This study is a step towards automatic video annotation of animals, where the bounding boxes are clustered on the go, and the animal identities are determined with minimal intervention from the user.

REFERENCES

[1] S. Schneider, G. W. Taylor, S. Linquist, and S. C. Kremer, "Past, present and future approaches using computer vision for animal re-identification from camera trap data," *Methods in Ecology and Evolution*, vol. 10, no. 4, pp. 461–470, 2019.

[2] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, vol. 1. Ieee, 2005.

[3] N. Dalal, B. Triggs, and C. Schmid, "Human detection using oriented histograms of flow and appearance," in *European conference on computer vision*. Springer, 2006.

[4] Q. Zhu, M.-C. Yeh, K.-T. Cheng, and S. Avidan, "Fast human detection using a cascade of histograms of oriented gradients," in *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*, vol. 2. IEEE, 2006.

[5] Z. Sun, G. Bebis, and R. Miller, "On-road vehicle detection: A review," *IEEE transactions on pattern analysis and machine intelligence*, vol. 28, no. 5, 2006.

[6] B. Coifman, D. Beymer, P. McLauchlan, and J. Malik, "A real-time computer vision system for vehicle tracking and traffic surveillance," *Transportation Research Part C: Emerging Technologies*, vol. 6, no. 4, pp. 271–288, 1998.

[7] X. Mei and H. Ling, "Robust visual tracking and vehicle classification via sparse representation," *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, vol. 33, no. 11, pp. 2259–2272, NOV 2011.

[8] W. Luo, J. Xing, A. Milan, X. Zhang, W. Liu, and T. K. Kim, "Multiple object tracking: A literature review," *Artificial Intelligence*, vol. 293, p. 103448, 2021. [Online]. Available: https://doi.org/10.1016/j.artint.2020.103448

[9] A. Milan, L. Leal-Taixé, I. Reid, S. Roth, and K. Schindler, "Mot16: A benchmark for multi-object tracking," *arXiv preprint arXiv:1603.00831*, 2016.

[10] Z. Shen, Z. Liu, J. Li, Y.-G. Jiang, Y. Chen, and X. Xue, "Dsod: Learning deeply supervised object detectors from scratch," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[11] Y. Bar-Shalom, "Tracking methods in a multitarget environment," *IEEE Transactions on automatic control*, vol. 23, no. 4, pp. 618–626, 1978.

[12] E. Meijering, O. Dzyubachyk, and I. Smal, "Methods for cell and particle tracking," *Methods in enzymology*, vol. 504, pp. 183–200, 2012.

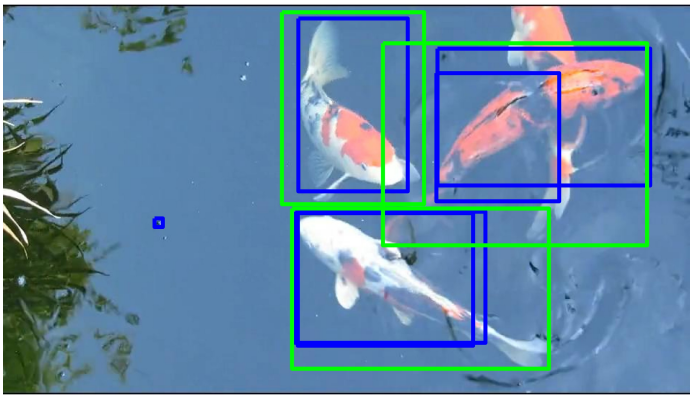
[13] K. Chen, J. Wang, J. Pang, Y. Cao, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Xu, Z. Zhang, D. Cheng, C. Zhu, T. Cheng, Q. Zhao, B. Li, X. Lu, R. Zhu, Y. Wu, J. Dai, J. Wang, J. Shi, W. Ouyang, C. C. Loy, and D. Lin, "MMDetection: Open mmlab detection toolbox and benchmark," *arXiv preprint arXiv:1906.07155*, 2019.

[14] Z. Wang, H. Zhao, Y.-L. Li, S. Wang, P. Torr, and L. Bertinetto, "Do different tracking tasks require different appearance models?" *Thirty-Fifth Conference on Neural Infromation Processing Systems*, 2021.

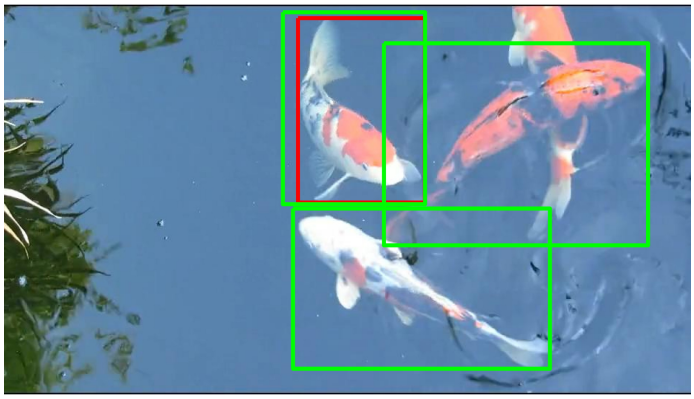
[15] R. Padilla, S. L. Netto, and E. A. B. da Silva, "A survey on performance metrics for object-detection algorithms," in *2020 International Conference on Systems, Signals and Image Processing (IWSSIP)*, 2020, pp. 237–242.

[16] R. Padilla, W. L. Passos, T. L. B. Dias, S. L. Netto, and E. A. B. da Silva, "A comparative analysis of object detection metrics with a companion open-source toolkit," *Electronics*, vol. 10, no. 3, 2021. [Online]. Available: https://www.mdpi.com/2079-9292/10/3/279

Frame # 497: Detector wins

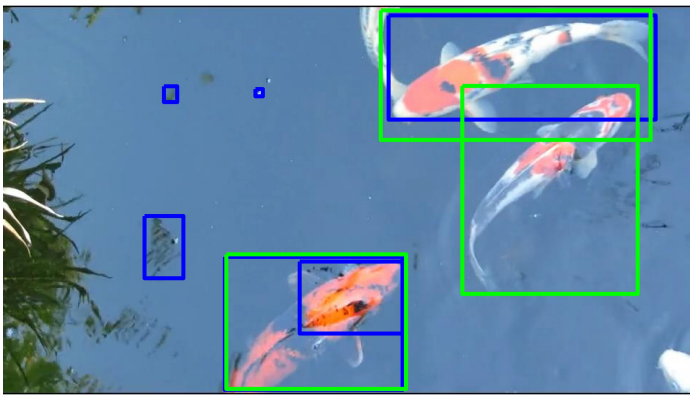


(a) Detector $AP = 100$ % (6 BB)

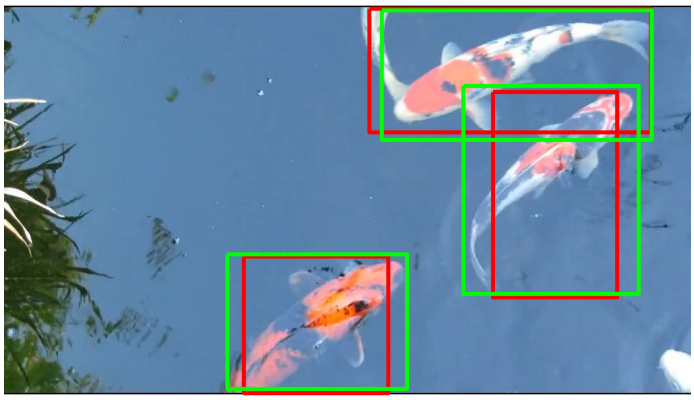


(b) Tracker $AP = 33.33$ % (1 BB)

Frame # 438: Tracker wins

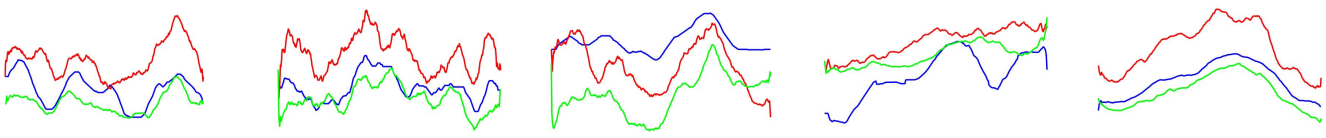


(c) Detector $AP = 27.78$ % (6 BB)



(d) Tracker $AP = 100$ % (3 BB)

Fig. 3: Example from the Koi fish video of differences in object detection between the Detector and the Tracker methods. The ground truth is shown with green, the Detector results, in blue, and the Tracker results, in red. In both images there are three ground truth bounding boxes.



(a) Koi fish (b) Pigeons (ground) (c) Pigeons (curb) (d) Pigeons (square) (e) Pigs

Fig. 4: The x-axis is the frame number and the y-axis is the number of bounding boxes per frame. The blue curve is the ground truth, the red is the detector output, and the green is the tracker output.

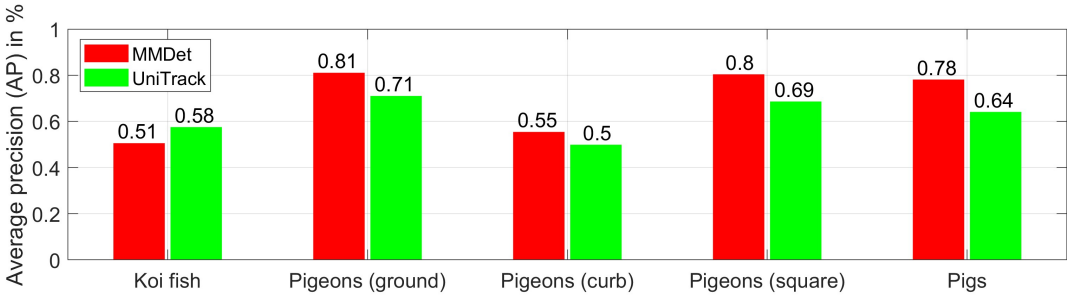


Fig. 5: Average Precision (AP) for MMDetection (MMDet) and UniTrack for the five videos.

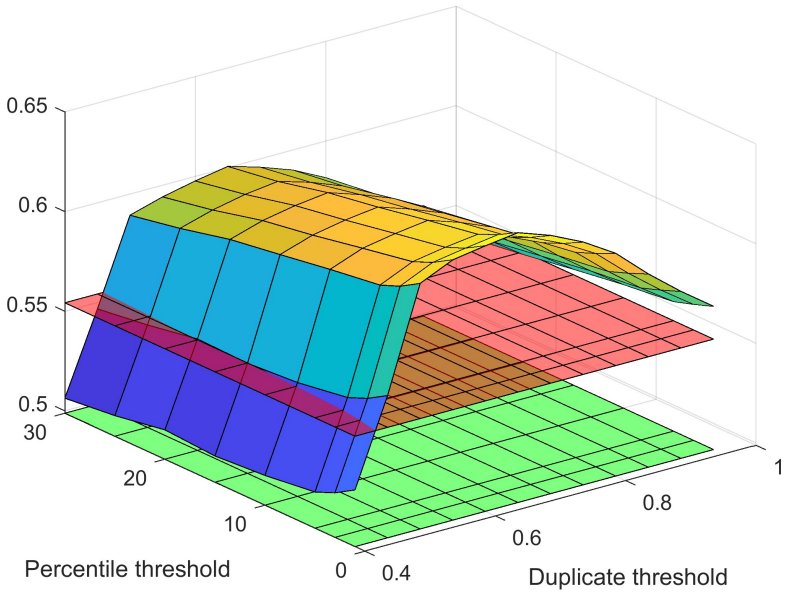
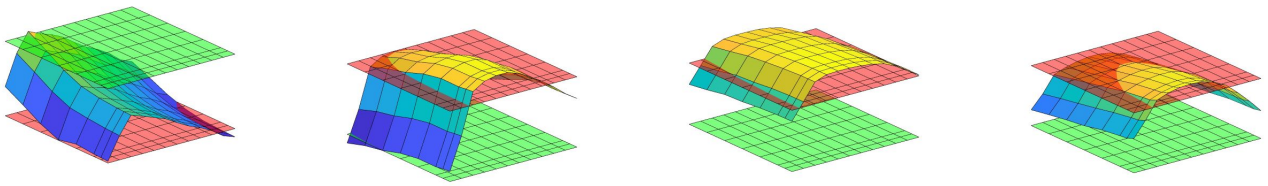


Fig. 6: Example of the AP obtained by the combination method in comparison with the AP of the detector (red plane) and the tracker (green plane) for the Pigeons (curb) video. The surface is drawn in the space of values spanned by the duplicate threshold D and the percentile threshold P.



(a) Koi fish (b) Pigeons (ground) (c) Pigeons (square) (e) Pigs

Fig. 7: AP surface obtained from the combination method for different parameter values P and D. The red plane is the constant AP value for the detector and the green plane is the constant AP value for the tracker.