A stacked meta-approach for object detection to reduce false positives in highly complex videos

Juan Manuel, Davila Delgado*
Birmingham City University, UK

Ari Yair, Barrera-Animas
Universidad Panamericana, México

ABSTRACT

False positives are a major problem when deploying object detection models in real-world conditions. Highly complex scenes are particularly difficult to process by standard object detection models. A novel meta-approach of stacked detection and the use of multiple frames to evaluate the preliminary detections is presented. The stacked approach leverages different types of architectures and performs multiple detections to reduce the number of false positives. The approach was qualitatively validated with videos taken from construction sites and compared with some of the most used architectures, i.e., Faster-RCNN and RetinaNet. Our approach can reduce the number of false positives and increase the detection accuracy.

CCS CONCEPTS

• **Machine learning algorithms, Ensemble methods**;

KEYWORDS

Computer Vision, Object detection, Synthetic approaches, Model Ensemble

ACM Reference Format:
Juan Manuel, Davila Delgado* and Ari Yair, Barrera-Animas. 2024. A stacked meta-approach for object detection to reduce false positives in highly complex videos. In *2024 the 8th International Conference on Innovation in Artificial Intelligence (ICIAI 2024), March 16–18, 2024, Tokyo, Japan.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3655497.3655501

1 INTRODUCTION

This study is part of a research project in collaboration with industry that focused on deploying machine learning (ML) object detection models in real-life construction sites. It was an 18-month collaboration in which camera and networking kits were deployed to two construction sites in the UK. Video recordings of the sites were collected for about 12-months which have been used to develop computer vision solutions for construction. One of the major problems identified in deploying object detection solutions for real-life applications is the false positive problem, which refers to the issue of incorrectly identifying an object that is not present in the image or misclassifying a region of the image that does not contain the object. This problem is critical for applications in which complex scenes are present, for example in which a multitude of varied objects are present in scenes, such as construction sites. Figure 1 presents an example of false positives in a real-world construction site using two of the most used detectors (Faster-RCNN and RetinaNet).

The objectives of the study presented in this paper were (1) to develop a so called "meta stacked approach" to reduce the number of false positives by leveraging the strengths of different object detection models and (2) to develop an algorithm that enables to reduce false positives in videos by leveraging the information of multiple frames in the detection. This meta-approach was used to enable the quick deployment of already tested object detection architectures and trained models to provide object detection services in real-world applications in construction.

The paper is structured as follows. Section 2 presents the background, Section 3 presents our proposed stacked meta-approach. Sections 4 and 5 presents the experiments and results. Section 6 presents the conclusions of the study.

2 BACKGROUND

2.1 Two-stage and one-stage detection models

ML approaches to detect objects carry out two subsequent tasks i.e., (1) find a number of objects of interest in the image and (2) classify those objects including the estimation of a bounding box in which the object is located. There are two main architectures to carry out those tasks, "two-stage" architectures separate the tasks in two stages, while "one-stage architectures" combine both tasks in a single step. The main two-stage and one-stage architectures are presented in Table 1. The generic two-stage architecture consists of two steps (1) regions in the images are defined, using either traditional computer vision methods or neural networks, in which relevant objects can potentially be located, then (2) the image features encountered in these regions are used for classification and bounding box estimation. In other words, the two- stage architecture first finds a region of interest and then uses this region for classification. Generally, two-stage architectures achieve high detection accuracy, but they are slower at inference due to the many steps required during the detection. The generic one-stage architecture performs the detection in all the image without the proposing relevant regions. This architecture is generally faster as less steps are required; however, accuracy is usually lower compared with two-stage architectures. Also, irregularly shaped objects or a group of small objects are not detected in a reliable manner. Note, that systematic comparisons among architectures is very difficult to achieve as the model parameters of the studies are not the same.

ICIAI 2024, March 16–18, 2024, Tokyo, Japan

ACM ISBN 979-8-4007-0930-2/24/03
https://doi.org/10.1145/3655497.3655501

Figure 1: Examples of false positives using FasterRCNN (left) and RetinaNet (right).

Table 1: Types of object detection architectures

Two Stage	Year	One Stage	Year
RCNN and SPPNET	2014	YOLO	2016
Fast RCNN and Faster RCNN	2015	SSD	2016
Mask R-CNN	2017	RetinaNet	2017
Pyramid Networks/FPN	2017	YOLOv3	2018
G-RCNN	2021	YOLOv4	2020
		YOLOR	2021
		YOLOv7	2022

2.2 Object detection models selected for the study

This subsection describes the two state-of-the-art detection algorithms used to evaluate the proposed stack approach to reduce the appearances of false detections of machinery in construction sites. Although it is out of the scope of this research to present a comprehensive benchmark of the performance of synthetic datasets generated in the broad state-of-the-art detection algorithms; the object detectors used were selected for their relevance, performance and wide use [1].

Faster R-CNN. Ren et al. in [2] introduced a Region Proposal Network (RPN) that trained together with Fast R-CNN for detection received the name of Faster R-CNN. Their work was centred on alleviating the computational bottleneck that was the region proposals on region-based convolutional neural networks (R-CNNs). To address this drawback, the authors proposed the use of RPN with a training scheme for Fast R-CNN that allows a training alternation between fine-tuning for the region proposal task and then fine-tuning for object detection. Moreover, the proposed algorithm does not need multi-scale- feature or -sliding windows to predict large regions due to the use of anchor boxes that are larger than the underlying receptive field; minimising execution time considerably. The authors evaluated their proposed detection algorithm on the PASCAL VOC 2007 and PASCAL VOC 2012 benchmarks using a onestage detection and a two-stage proposal plus detection approaches. Results showed that Fast R-CNN detectors trained with RPNs achieved better mean Average Precision (mAP) values than when trained with Selective Search; as well as that the one-stage system obtained a lower mAP and was slower than the two-stage system.

RetinaNet. Later, in 2017, Lin et al. [3] demonstrated that one-stage detectors could be faster, simpler and can obtain better accuracy than two-stage detectors by addressing the extreme foreground-background class imbalance problem during training of dense detectors. To address this issue, the authors proposed a Focal Loss function that reshapes the standard cross-entropy loss so that it down-weights the contribution of easy examples during training and quickly focus the model on difficult examples. A one-stage detector, that features an efficient in-network feature pyramid as a backbone network that uses anchor boxes, called RetinaNet was designed and trained to evaluate its proposed approach. The evaluation results demonstrate that RetinaNet trained with Focal Loss achieved a similar speed of state-of-the-art one-stage detectors and better values in the COCO AP metric than four two-stage detectors based on Faster R-CNN and 3 one-stage detectors.

3 STACKED META APPROACH

This subsection describes the proposed approach of building a stack of object detectors to both, retain the majority number of detections in an image and to decrease the number of false positives displayed as a processed image.

The stacked approach is comprised of two different object detectors trained with the same dataset. The two detectors must be chosen by the researcher accordingly to the characteristics of each one in accordance to the problem to be addressed. However, the suggestion is to select two detectors that can act as complementary to each other; for example, one that is good at recognising small

objects and other that inference time is smaller. It is out of the scope of this research work to provide any comparison or combinatorial analysis regarding the arrangement of the object detection algorithms. The general stacked approach, presented in Algorithm 1, is as follows:

- Select the object detection algorithm that provides the major range of detections as the first detector.
- Set a comprehensible low detection threshold as a base detection for the previously selected detector. For example, between 70% and 89%. Set a stricter detection threshold that will be used by both detectors as a rectification process.
- Perform object detection over an image with the first detector.
- For every detection resulted from the previous step, perform object detection with both detectors configured with the stricter detection threshold as reification process.
- Only compare the results of both detectors at the last step. If one of the detectors was unable to detect an object, then discard this detection. On the other hand, retain the detection made by the second detector.
- If the classes of both detections are the same; otherwise, retain both detections.

Moreover, in order to extent the proposed approach when dealing with live-streams, it is proposed a verification process as a wrap layer to the stack detection approach. The aim of the verification step is to provide an extra layer to deal with objects that promptly appears or quickly moves during consecutive images. Therefore, it is proposed as "addon-step" to retain objects that are expected to been visible inside camera's view while moving around without presenting a disappearance/appearance behaviour.

The general workflow, presented in Algorithm 2, is as follows:

- Set a number of consecutive frames/images detections that will be used in the verification. The frames/images are only those in which the detection algorithm, in this case the stacked approach, is performed.
- Perform object detection (stacked detection approach) in each frame/image (were corresponding).
- If the number of frames/images being processed is equal to the previously established number, execute the verification process.
- For the verification process, match the bounding boxes of the different detections retained in each of the processed frame/image.
- Return/show only those detections that persist in the processed frames/images.

4 EXPERIMENTS

In this section it is described the parameters and configurations used to perform the test of the proposed stack detection approach and the verification through time procedure. Firstly, the dataset and the object detection algorithms used are provided. Later, the parameters of the stack detection approach are detailed. Lastly, the evaluation procedure is presented.

Algorithm 1 Stacked meta-approach for single frames

Data: datasetM ← Dataset of construction machinery
firstDetector ← First object detector option trained with datasetM
secondDetector ← Second object detector option trained with datasetM baseDetectionThreshold ← $Z \in N^{89}{}_0$
strictDetectionThreshold ← $Z \in N^{100}{}_{90}$ frame ← Input image
Result: Detections of construction machinery
firstDetectorOne ← firstDetector(detectionThreshold = baseDetectionThreshold) firstDetectorTwo ← firstDetector(detectionThreshold = strictDetectionThreshold) secondDetector ← secondDetector(detectionThreshold = strictDetectionThreshold) baseDetections ← firstDetectorOne(frame) agreedDetections ← {}
for detection in baseDetections **do**
maskedFrame ← MaskImage(frame); /* Hide everything in the frame with exception of the current detection */
firstDetections ← firstDetectorOne(maskedFrame)
singleFirstDetection ← SingleDetection(firstDetections) ;/* Iterate thought the detections to only retrieve the detection with the biggest bounding box */
secondDetections ← secondDetector(maskedFrame)
singleSecondDetection ← SingleDetection(secondDetections) **if** singleFirstDetection ∧ singleSecondDetection$\not=$ ∅ **then if** singleFirstDetectionobjectClass = singleSecondDetectionobjectClass
then agreedInferenceScore ← Average(singleFirstDetectioninferenceScore,singleSecondDetectioninferenceScore) agreedDetections ← agreedDetections ∪{(objectClass,agreedInferenceScore,singleSecondDetectionboundingBox)}
if singleFirstDetectionobjectClass$\not=$ singleSecondDetectionobjectClass then detectionClasses ← (singleFirstDetectionobjectClass,singleSecondDetectionobjectClass) detectionScores ← (singleFirstDetectioninferenceScore,singleSecondDetectioninferenceScore) agreedDetections ← agreedDetections ∪ {(detectionClasses,detectionScores,singleSecondDetectionboundingBox)}

4.1 Dataset and training process

For training purposes, a synthetic dataset were generated following the approach described in [9].

The characteristics of the generated synthetic dataset used for training the object detection algorithms is summarised in Table 2 and Table 3.

Subsequently, the previously generated synthetic dataset was used to train two object detection algorithms using the Detectron2 [4] platform, Faster R-CNN and RetinaNet.

The training parameters used to train both detectors were the same as described in [9]. Both object detection algorithms also used the MS COCO dataset as transfer learning approach.

Table 4 and Figure 2 show the results of the training process of the Faster R-CNN and RetinaNet detection algorithms trained with the generated synthetic dataset. The evaluation of the training performance of the detectors was performed using the Detectron2 library COCO Evaluator [5]. Detectron2 uses the COCO's metrics

Table 2: Synthetic dataset summary

Characteristic	Value
Total number of images	13,553
No. of training images	13,347
No. of validation images	206
No. of Categories	7
Categories names	Excavator, Bulldozer, Truck, Roller, Dump Truck, Loader, Concrete Mixer

Table 3: Category distribution of the synthetic dataset

Category	Training Set	Validation Set
Excavator	4,854	69
Bulldozer	6,247	89
Truck	5,634	100
Roller	5,019	72
Dump Truck	5,662	89
Loader	4,472	70
Concrete Mixer	7,054	95
Total	3,8942	584

Algorithm 2 Verification through time for multiple frames

Data: proposedDetectorEnsemble ← Ensemble of object detectors (Algorithm 1) skipFramesNumber ←Z∈N > 0 boundingBoxDifferencePercentageThreshold ← Z∈N1000 frames ← Input video frames
Result: Detections of construction machinery
for frame in frames **do**
  detectionsToBeCompared ← {}
  if frame_number == skipFramesNumber **then**
    detections ← proposedStackDetector (frame)
    detectionsToBeCompared ← detectionsToBeCompared {detections}
  else
    detections ← proposedStackDetector(frame)
    detectionsThroughTime ← {}
    for detection in detections **do**
detectionsHomonyms ← 0
for previousDetections in detectionsToBeCompared **do**
  boundingBoxDifference ← DifferencePercentange(detection,previousDetection)
if boundingBoxDifference ≤ boundingBoxDiffferencePercentageThreshold
      then detectionHomonyms ← detectionHomonyms 1
      if detectionHomonyms ≥ skipFramesNumber
then detectionsThroughTime ← detectionsThroughTime {detection}
Display(detectionsThroughTime)

[6] of Average Precision (AP), AP50, AP75, APsmall, APmedium, and APlarge.

Table 4: Object detectors performance

COCO eval. metrics	Faster R-CNN	RetinaNet
AP (%)	83.93	87.86
AP50 (%)	98.69	97.07
AP75 (%)	91.81	93.43
APsmall (%)	58.73	57.45
APmedium (%)	85.04	90.44
APlarge (%)	95.64	97.56

4.2 Parameters of the stack and verification approach

The parameters and components of the stack detection approach and the verification through time proposed process are concentrated in Table 5.

4.3 Evaluation process

To evaluate the proposed stacked detection meta approach, one experiment was carried out. A set of 3 videos corresponding to different construction site scenes that display real-world construction site scenarios were selected for testing. This selection aims to pursuit two objectives; first, to test the proposed approach in a real-world scenario; and second, to provide insights on the performance of object detectors trained with synthetic images and tested with real-world scenes. Image frames from videos of the scenes were selected and were processed with the proposed stacked meta-approach of object detectors (described in Algorithm 1). The Faster R-CNN (described in subsection 4.1) with testing thresholds of 70% and 90%, as well as with the RetinaNet (described in subsection 4.1) with a testing threshold of 90%. Finally, the processed images were

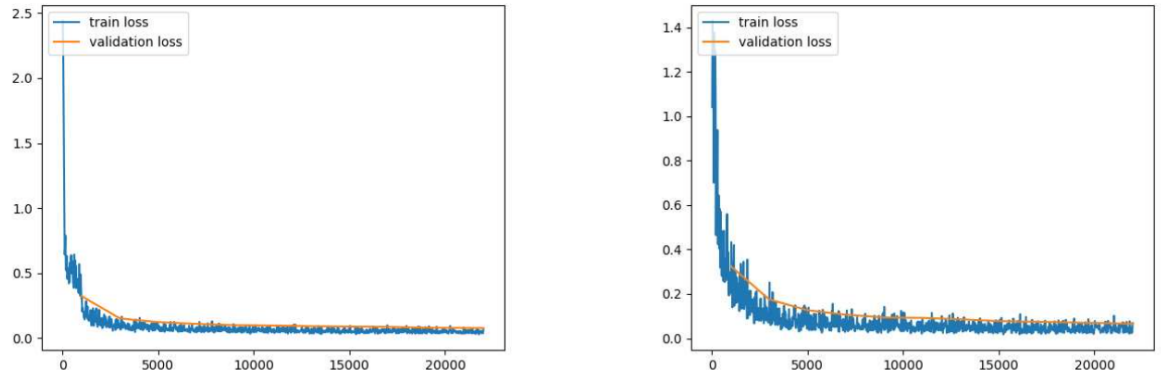

Figure 2: Object detectors training and validation loss curves

Table 5: Parameters and components of the proposed approach

Parameter or component	Value
Stack detection	Synthetic dataset as described in 4.1
Dataset	
First detector	Faster R-CNN as described in 4.1
Second detector	RetinaNet as described in 4.1
Base detection threshold	70%
Strict detection threshold	90%
Verification through time	25%
Bounding box difference percentage	
Number of images before verification	2

visually compared to review the performance of the proposed stack approach regarding the appearance of false positives.

Also, for this experiment, a video object detection script was developed with the following characteristics: tracking for 15 frames with CSRT tracker [7] provided in OpenCV (Open Source Computer Vision) Library [8] version 3.4.14, and three independent different detectors. The first detector correspond to the proposed stack, the second was the Faster R-CNN, and the third one was the RetinaNet. All detectors were wrapped as described in the verification through time process previously described in Algorithm 2. The results of using the three different object detection approaches were visually compared. A Windows system with 8 logical processors Intel Core i7-7700K CPU @ 4.20 GHz, 32 GB of RAM, and 2 NVIDIA GeForce GTX 1080 Ti graphic cards was used to perform all the experiments.

5 RESULTS

This section presents the results of the experiments carried out using the stacked meta approach, the verification through time process, and the independent detectors of Faster R-CNN and RetinaNet. Note that this is a qualitative evaluation of the results. A systematic quantitative evaluation is planned for future work for which a testing dataset is being prepared and labelled manually. Figure 3 presents selected frames from three videos of construction sites, for each frame the detection was carried out using Faster-RCNN, RetinaNet, and our approach. It can be noted that for all cases our presents a lower number of false positives than the other two approaches, and it is able to detect more objects correctly as well. RetinaNet has less false positives than Faster-RCNN, but it fails to detect relevant objects, e.g., workers.

6 CONCLUSIONS

This paper presents two novel approaches to address the high number a false positives encountered in object detection models when deploy in real-life conditions and in very complex scenes such as construction sites. A meta-approach that leverages two different object detection models, a one-stage detector and a two-stage detector was developed. This approach takes advantage of the benefits of the different types of architectures to reduce the number of false positives. This approach was complemented with an additional algorithm that makes use of multiple video frames to improve the detection further and eliminate more false positives. The approach was validated testing the approach with real-world videos taken at construction sites and compared with two of the most used architecture models for object detection. The qualitative validation shows that our presented approach reduces the number of false positives and increases the accuracy of the detection.

ACKNOWLEDGMENTS

The authors would like to express their sincere gratitude to Innovate UK (Grant Application No 85549) for providing the financial support for this study.

a b c

Figure 3: Results of the experiments (a) Faster-RCNN, (b) RetinaNet, and (c) stacked meta-approach with time verification.

REFERENCES

[1] A. Bochkovskiy, C.-Y. Wang, H.-Y. M. Liao, Yolov4: Optimal speed and accuracy of object detection (2020). arXiv:2004.10934.

[2] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: Towards real-time object detection with region proposal networks, Advances in neural information processing systems 28 (2015) 91–99.

[3] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Doll´ar, Focal loss for dense object detection, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 2980–2988.

[4] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, R. Girshick, Detectron2, https://github.com/facebookresearch/detectron2 (2019).

[5] Detectron2 contributors, Detectron2's documentation, https://detectron2.readthedocs.io/en/latest/ (2019-2020).

[6] COCO, COCO Detection Evaluation, https://cocodataset.org/detection-eval (2015).

[7] doxygen, The OpenCV Reference Manual, 1st Edition (April 2021).

[8] G. Bradski, The OpenCV Library, Dr. Dobb's Journal of Software Tools (2000).

[9] A. Y. Barrera-Animas, J. M. Davila Delgado, Generating real-world-like labelled synthetic datasets for construction site applications, Automation in Construction, 151, July, (2023), 104850.