EXPLORING SEIDEN: SAMPLING WITH RARE EVENTS FOR INDEXING IN VIDEO DATABASE SYSTEMS

A Thesis

By

Cameron Potter

Approved by:

Dr. Joy Arulraj, Advisor
College of Computing
Georgia Institute of Technology

Date Approved: 5/2/2023 | 4:05 PM EDT

DocuSigned by:
Joy Arulraj
A83AB59BC4E741D...

Dr. Chao Zhang, Second Reader
College of Computing
Georgia Institute of Technology

Date Approved: 5/1/2023 | 4:58 PM EDT

DocuSigned by:
Chao Zhang
5D656B40A557434...

TABLE OF CONTENTS

LIST OF FIGURES

SUMMARY

SEIDEN is a video database management system (VDBMS) that utilizes lightweight oracle models like YOLOv5 to derive labels and generate proxy scores for efficient video query processing. To promote optimization for rare events, we analyze the relationship between anchor count, DNN invocations, and recall for specific frame ratios and object classes. The performance of SEIDEN is evaluated based on precision, recall, and F1 score using these parameters.

The results demonstrate that SEIDEN's performance is significantly affected by the frame ratio and DNN invocation values used, with interesting patterns emerging for common and rare object classes. Increasing the DNN invocation value improves recall for both classes but at the cost of decreased precision. However, the trend is more significant for the rare class, suggesting that allocating more computational resources could enhance SEIDEN's performance more substantially for rare events than for common events.

The results yield insights into the optimization of SEIDEN's performance in object detection tasks. The findings further demonstrate the potential of SEIDEN as a fast and accurate VDBMS, even for rare events. Future research could expand on these findings by examining SEIDEN's generalizability to other datasets and comparing its performance with other methods.

CHAPTER 1

INTRODUCTION AND BACKGROUND

Video data is increasingly prevalent in our daily lives, from security cameras and traffic monitoring to social media and entertainment. Video databases have become an important tool for managing and analyzing large collections of videos. However, as the amount of video data grows, traditional methods for querying databases become inefficient, especially when using computer vision models for object detection tasks. To address this issue, researchers have developed video database management systems (VDBMSs) that optimize query processing to provide faster and more accurate results [1, 2, 3].

VDBMSs can process queries that are either aggregate or retrieval based. Aggregate queries use aggregate functions, such as SUM or COUNT, to return a single value that summarizes a group of frames. For example, an aggregate query could return the total number of cars in a video dataset. Retrieval queries, on the other hand, retrieve all frames that contain a specific object of interest. These queries are useful for applications such as video surveillance or object tracking. Here, we focus on retrieval queries.

To speed up these queries, previous VDBMSs utilized proxy models which sacrifice accuracy for speed compared to the oracle model. These proxy models are typically simplified deep learning models like RESNET-18 [4, 5]. Several state-of-the-art VDBMSs still use MASK R-CNN as the reference model [1, 2, 6]; however, recent advancements in computer vision have challenged the notion that the oracle model is always slower. For instance, YOLOv5 is a new object detection model that outperforms MASK R-CNN in terms of both speed and accuracy on the COCO dataset [7]. The inference speed of YOLOv5 is 11.2 times faster than MASK R-CNN and almost as fast as RESNET-18. This means there's no longer a need for a proxy model to pre-process frames before sending them to the oracle model.

One method for these tasks is TASTI which creates a proxy model for the given video dataset and MASK R-CNN oracle model to create an index of significant frames (e.g. 5% of all the frames)

[6]. They use an farthest-point-first (FPF) algorithm based on the RESNET-18 features to pick the representative frames and only run the MASK R-CNN oracle model on these frames during query exectution. This results in proxy scores of all the frames in the video through the RESNET-18 model. The use of RESNET-18 as a proxy model in TASTI is no longer justifiable as its inference time is now comparable to recently released YOLOv5s models. This means that the query processing time and accuracy drop associated with TASTI is no longer necessary. Replacing RESNET-18 with an even lighter deep learning model or using cheaper features like pixel values would result in a greater loss in accuracy.

SEIDEN is a novel VDBMS developed by the Georgia Tech Database Group that processes queries efficiently without relying heavily on compressed deep learning models. It takes advantage of the smooth and continuous motion of video frames by considering only every k^{th} frame from the video, assuming that neighboring frames do not add much additional information. To select representative frames, SEIDEN uses a combination of frame sampling and exploration-exploitation strategies, exploiting label differences and exploring unsampled segments of the video that are more likely to provide useful information. For answering both aggregate and retrieval queries, SEIDEN utilizes lightweight oracle models such as Yolov5s to derive labels and generate proxy scores.

While SEIDEN has demonstrated significant improvements over other methods, it may not perform optimally for rarer events in retrieval queries, such as a bus in an intersection, because the exploration-exploitation strategies may not select optimal frames. For instance, if a bus only appears between two frames that were sampled, the algorithm would assume that there is no bus in between those frames. On the other hand, if the algorithm selects one frame with a bus in the initial steps, it may incorrectly weight all frames in between the other sampled frames. Thus, more optimized sampling methods are required for rarer events.

Retrieval queries for rare events present a particular challenge due to the higher probability of false negatives, potentially resulting in low recall. Low precision is also a possibility when sampling frames, especially for rare events, leading to incorrectly weighted neighborhoods of frames. To address these issues, we explore the relationship between anchor budget, DNN invocations, and recall for specific object classes, and optimize SEIDEN's performance in object detection tasks, focusing on rare events. Specifically, we will investigate how adjusting the number of initial frames sampled and the maximum number of new frames explored affects recall, and provide insights into SEIDEN's strengths and weaknesses for specific object classes. Our study builds upon previous work on SEIDEN and aims to enhance its effectiveness in challenging scenarios.

CHAPTER 2
METHODOLOGY

We seek to improve the performance through optimizing the parameters of SEIDEN. Specifically, we can vary the number of initial frames sampled, the anchor budget L, and the maximum number of new frames explored, the additional anchor frames Q, for consistent results. The goal is to optimize Q, also known as DNN invocations, for specific recall with varying L. Initially, the goal was to optimize L for some desired maximum run time; however, run time can be inconsistent on different machines and is not ideal to measure cost. TASTI demonstrated their target labeler invocations, similar to our Q, was the primary cost in index construction [6]. Hence, we can attempt to optimize L and Q for consistent metrics, e.g. recall. We then can evaluate our methods using the UA-DETRAC dataset [8].

The UA-DETRAC dataset is a collection of short videos recorded by traffic cameras, each with a duration of one minute. The dataset originally consisted of several independent videos, but the authors combined 22 of these videos resulting in a new dataset that simulates shot changes which are common in movies. The resolution of the dataset is 960×540 with 3 color channels, and it contains a total of $N = 83791$ frames. Many of the videos in this dataset depict dense traffic scenes, resulting in a high occupancy rate of 89.33% and an average object count of 2.99 for the 'car' event.

One of the main concerns with rare events is the higher probability of the initial frame selection missing the frames containing the rare event, which can result in a lack of information to base further frame sampling on. This can lead to most of the rare class not being seen, resulting in a large number of false negatives and low recall. To address this, we can utilize a SUPG precision query [6], where the user can specify a target precision and return frames containing the class. This allows us to base our experiments on optimizing recall for rare events, which is particularly important in applications such as traffic monitoring where accidents or traffic violations are of great interest.

To evaluate the performance of SEIDEN in detecting rare events, we selected the 'bus' category as our rare event since it appeared less than 2% as frequently as the 'car' category in the UA-DETRAC dataset. Specifically, the Yolov5s model detected ‘bus’ only 2,167 times out of the total 83,791 frames, whereas ‘car’ was detected 121,358 times. This significant difference in frequency makes ‘bus’ an ideal candidate for evaluating SEIDEN’s ability to detect rare events.

To optimize the performance of SEIDEN, we conducted experiments using different values of L and Q. We evaluated all values of L in the set $\{\frac{n}{20} \cdot N \mid n \in \mathbb{N}, 1 \leq n \leq 20\}$, where N is the total number of frames in the UA-DETRAC dataset, and all values of Q in the set $\{n \cdot 200 \mid n \in \mathbb{N}, 1 \leq n \leq 25\}$. The range of values was chosen to provide a comprehensive view of the relationship between Q and L. For each combination of Q and L, we recorded the precision, recall, and F1 score. We then performed a grid search to identify the optimal values of Q and L for a specific recall bound. We repeated each experiment three times and calculated the average precision, recall, and F1 score to minimize the effects of outliers and random fluctuations in the algorithm’s performance.

To evaluate the performance of our proposed method, we use YOLOv5 as the reference inference model unless stated otherwise. We compare the labels assigned by our algorithm to those assigned by YOLOv5 for each frame in the video and rank them accordingly. We repeat the evaluation process for each combination of L and Q values used in our experiments. Finally, we analyze the results to determine the optimal values for L and Q to achieve the desired recall for the two events.

CHAPTER 3
RESULTS & ANALYSIS

After analyzing the results for varying frame ratios and DNN invocation values, we created color maps to visually represent the recall, precision, and F1 score for the car and bus classes. The color maps illustrate the relationship between the values of these metrics and the two variables.

In Figure 3.1, we present the recall results for car and bus, where brighter colors indicate larger values. Notably, for lower frame ratios, bus generally outperformed car with respect to recall values (≤ 0.4), while car had higher recall for middle frame ratios (between 0.45 and 0.6).

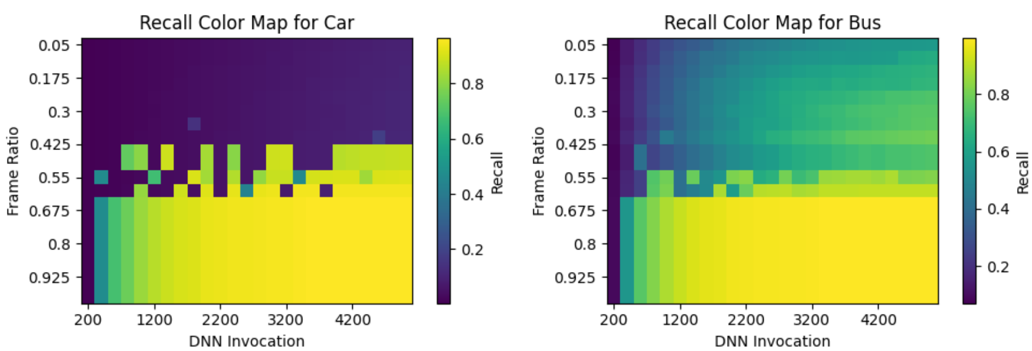

Figure 3.1: Color map of recall results with frame ratio and DNN invocation for the ‘car’ (left) and ‘bus’ (right) classes.

Moreover, we observed that recall generally increased with an increase in DNN invocation values. This trend was more pronounced for bus, with a visible, steady increase in recall for all frame ratios. However, the trend was still present for car. Bus had higher recall than car for 89% of the results, with car performing better for only about half of the results when the frame ratio was between 0.45 and 0.6. This observation goes against the intuition that bus, being a rarer class, would naturally have lower recall. Therefore, it implies that the model is either missing more true positives or producing more false negatives than expected for car.

In Figure 3.2, we can see the precision results for car and bus. It shows that precision generally decreases as the DNN invocation increases. This trend is more prominent for the bus class, with visible, steady decreases for all frame ratios, but the trend is still slightly present for the car class. For precision, the car class generally performed better than the bus class, with the bus class only outperforming the car class in 5.8% of tests. It's worth noting that in nearly all cases where the bus class had better precision, the frame ratios and DNN invocations were low. This is intuitive because lower frame ratios and DNN invocations are less likely to produce false positives for rarer events due to limited information about their locations in the video from initial sampling and limited exploration abilities.

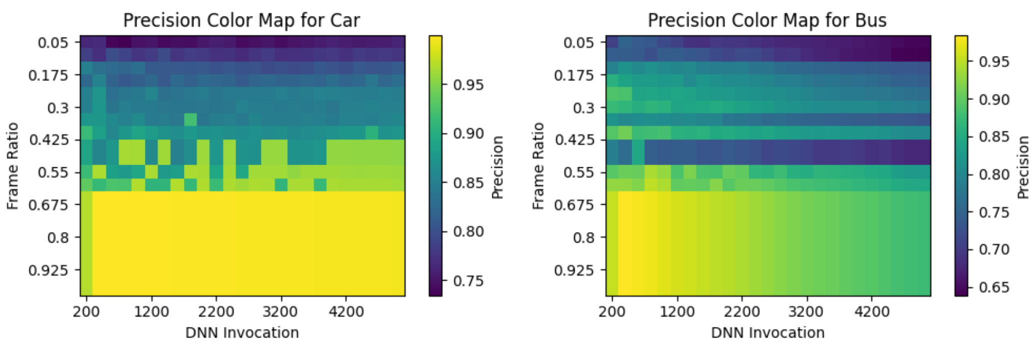

Figure 3.2: Color map of precision results with frame ratio and DNN invocation for the 'car' (left) and 'bus' (right) classes.

In Figure 3.3, we present the F1 score results for car and bus which illustrates that F1 score generally increases as DNN invocation increases. This trend is more prominent for bus, but is still slightly present for car. Bus had higher F1 scores in 63.2% of tests, with most of these occurring at lower frame ratios or lower DNN invocations. However, for larger frame ratios and DNN invocations, there is a greater chance of car achieving better F1 scores, likely due to the increasing information available and the already high average object count of 2.99 for car in the dataset.

A noteworthy finding from our analysis is that precision, recall, and F1 score did not change as the frame ratio increased beyond 0.65. This lack of change can be attributed to the fact that frame sampling did not provide additional information when most frames had already been sampled. This is an important observation because SEIDEN assumes that frames near ones known to contain an event are also likely to contain the same event due to continuity of motion. Once enough frames have been examined, there may not be enough frames in between to take advantage of this assumption, which

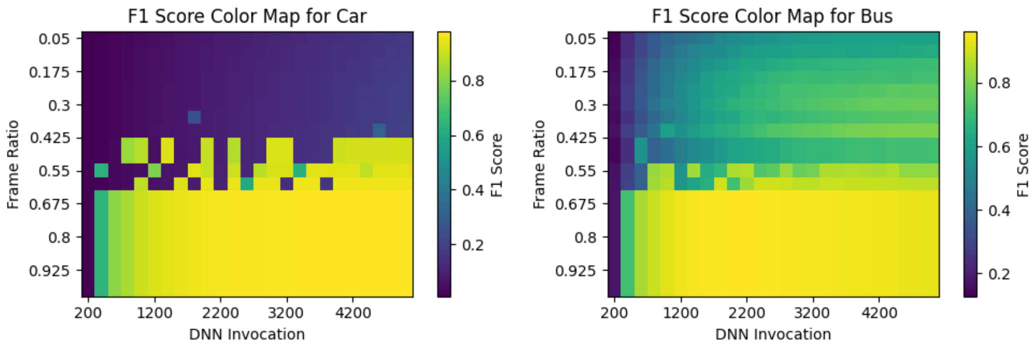

Figure 3.3: Color map of F1 score results with frame ratio and DNN invocation for the 'car' (left) and 'bus' (right) classes.

could explain the frame ratio limit we observed.

Returning to recall, we can observe the minimal DNN invocations required to achieve a recall lower bound and frame ratio for car and bus in Figure 3.4 and Figure 3.5 respectively. For some combinations, there was no DNN invocation that could reach the lower bound, and these points are indicated with "—". Approximately 61.2% of all tests were able to reach the recall lower bound for some DNN invocation for car, while for bus, this number was approximately 81.7%. Bus generally required lower DNN invocations than car, likely due to the already higher recall values for the class. As expected, the minimal DNN invocations tended to increase as the recall lower bound increased, and they decreased as the frame ratio increased.

Overall, these results provide valuable insights into the performance of SEIDEN and the behavior of the car and bus classes in the dataset.

Frame Ratio	Recall Lower Bound for Car																		
	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5	0.55	0.6	0.65	0.7	0.75	0.8	0.85	0.9	0.95
0.05	2800	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
0.1	2600	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
0.15	2400	5000	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
0.2	2400	4800	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
0.25	2400	4800	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
0.3	2400	4800	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
0.35	1800	1800	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
0.4	2400	4600	4600	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
0.45	800	800	800	800	800	800	800	800	800	800	800	800	800	800	1000	1400	1400	—	—
0.5	800	800	800	800	800	800	800	800	800	800	800	800	800	800	1000	1400	1400	—	—
0.55	400	400	400	400	400	400	400	400	400	1200	1200	1200	1200	1800	1800	1800	1800	1800	—
0.6	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1600	2000	—
0.65	400	400	400	400	400	400	400	400	400	600	600	600	600	800	1000	1000	1200	1600	3000
0.7	400	400	400	400	400	400	400	400	400	600	600	600	600	800	1000	1000	1200	1600	3000
0.75	400	400	400	400	400	400	400	400	400	600	600	600	600	800	1000	1000	1200	1600	3000
0.8	400	400	400	400	400	400	400	400	400	600	600	600	600	800	1000	1000	1200	1600	3000
0.85	400	400	400	400	400	400	400	400	400	600	600	600	600	800	1000	1000	1200	1600	3000
0.9	400	400	400	400	400	400	400	400	400	600	600	600	600	800	1000	1000	1200	1600	3000
0.95	400	400	400	400	400	400	400	400	400	600	600	600	600	800	1000	1000	1200	1600	3000
1.0	400	400	400	400	400	400	400	400	400	600	600	600	600	800	1000	1000	1200	1600	3000

Figure 3.4: Table giving the smallest DNN Invocation for car given recall lower bounds and frame ratio.

Frame Ratio	Recall Lower Bound for Bus																		
	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5	0.55	0.6	0.65	0.7	0.75	0.8	0.85	0.9	0.95
0.05	200	400	600	800	1000	1200	1600	1800	2400	3000	4000	—	—	—	—	—	—	—	—
0.1	200	400	600	800	1000	1200	1400	1600	2000	2400	3000	4600	—	—	—	—	—	—	—
0.15	200	400	400	600	800	1000	1200	1600	1800	2200	2600	3400	4600	—	—	—	—	—	—
0.2	200	400	400	600	800	1000	1200	1400	1600	2000	2400	2800	4200	—	—	—	—	—	—
0.25	200	400	400	600	800	1000	1200	1400	1600	2000	2200	2600	3200	4000	—	—	—	—	—
0.3	200	400	400	600	800	1000	1200	1400	1600	1800	2200	2600	3200	3800	4800	—	—	—	—
0.35	200	400	600	600	800	1000	1200	1400	1600	1800	2200	2400	3000	3800	4800	—	—	—	—
0.4	200	400	400	600	800	800	1000	1000	1000	1800	2200	2400	2600	3200	3800	—	—	—	—
0.45	200	400	600	600	600	600	600	600	1800	2000	2400	2800	3200	3800	—	—	—	—	—
0.5	200	400	600	600	600	600	600	600	1800	2000	2400	2800	3200	3800	—	—	—	—	—
0.55	200	400	400	600	800	800	800	800	800	800	800	800	800	800	1000	2000	3400	—	—
0.6	200	400	400	600	800	800	800	800	800	800	800	800	800	800	800	1000	1800	2800	—
0.65	200	400	400	400	400	400	400	400	400	400	400	600	600	600	600	800	1000	1200	1600
0.7	200	400	400	400	400	400	400	400	400	400	400	600	600	600	600	800	1000	1200	1600
0.75	200	400	400	400	400	400	400	400	400	400	400	600	600	600	600	800	1000	1200	1600
0.8	200	400	400	400	400	400	400	400	400	400	400	600	600	600	600	800	1000	1200	1600
0.85	200	400	400	400	400	400	400	400	400	400	400	600	600	600	600	800	1000	1200	1600
0.9	200	400	400	400	400	400	400	400	400	400	400	600	600	600	600	800	1000	1200	1600
0.95	200	400	400	400	400	400	400	400	400	400	400	600	600	600	600	800	1000	1200	1600
1.0	200	400	400	400	400	400	400	400	400	400	400	600	600	600	600	800	1000	1200	1600

Figure 3.5: Table giving the smallest DNN Invocation for bus given recall lower bounds and frame ratio.

CHAPTER 4
DISCUSSION & CONCLUSION

Here, we explored the use of SEIDEN, a fast, innovative, and straightforward VDBMS developed by the Georgia Tech Database Group. The goal was to optimize the performance of SEIDEN by analyzing the relationship between anchor budget L, DNN invocations Q, and recall. Specifically, we focused on optimizing DNN invocations for specific recall bounds with varying frame ratios for common (car) and rare (bus) events. To evaluate the performance of SEIDEN, we also measured precision and F1 score using these parameters. Our analysis revealed that SEIDEN's performance is significantly affected by the frame ratio and DNN invocation values used, and interesting patterns emerged in the model's performance for the car and bus classes.

We conducted a grid search optimization to determine the optimal values of DNN invocations for specific recall bounds and frame ratio values. The results showed that SEIDEN had a higher recall and F1 score for the bus class, but lower precision compared to the car class. This implies that the model is better at detecting instances of the bus class but may produce more false positives. Conversely, the car class had higher precision, indicating that the model performs better at accurately identifying instances of the car class while returning fewer instances. It's worth noting that increasing the DNN invocation value improved the recall for both classes, while the precision decreased. However, the trends were more significant for the bus class, suggesting that allocating more computational resources could enhance the model's performance more substantially for the bus class than the car class.

Despite our efforts, there are limitations that need to be considered. We focused only on one common and one rare class, car and bus respectively, and it is possible that our results could differ with other class choices. Further research is needed to explore the relationship between rarity and optimal parameters in more detail. Furthermore, due to computational constraints, we could only evaluate SEIDEN's performance on the UA-DETRAC dataset [8] and a SUPG precision query [6]. While this dataset provided diverse examples to test and the precision query was chosen to represent a wide range of retrieval queries, future studies should examine the generalizability of our findings to other datasets and queries. In addition, to fully understand SEIDEN's strengths and weaknesses, it would be useful to compare its performance with that of other methods.

In conclusion, we highlight the importance of selecting appropriate values for the frame ratio and DNN invocation to optimize the performance of SEIDEN in object detection tasks. Our results also provide insights into the model's performance for specific object classes, particularly for the rare class. Overall, our findings further suggest that SEIDEN is a valuable tool for improving the speed and accuracy of object detection in videos, even for rare events.

REFERENCES

[1] D. Kang, P. Bailis, and M. Zaharia, "Blazeit: Fast exploratory video queries using neural networks," *CoRR*, vol. abs/1805.01046, 2018. arXiv: 1805.01046.

[2] D. Kang, J. Emmons, F. Abuzaid, P. Bailis, and M. Zaharia, "Optimizing deep cnn-based queries over video streams at scale," *CoRR*, vol. abs/1703.02529, 2017. arXiv: 1703.02529.

[3] Y. Lu, A. Chowdhery, S. Kandula, and S. Chaudhuri, "Accelerating machine learning inference with probabilistic predicates," in *Proceedings of the 2018 International Conference on Management of Data*, ser. SIGMOD '18, Houston, TX, USA: Association for Computing Machinery, 2018, pp. 1493–1508, ISBN: 9781450347037.

[4] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[5] W. Liu *et al.*, "SSD: single shot multibox detector," *CoRR*, vol. abs/1512.02325, 2015. arXiv: 1512.02325.

[6] D. Kang, J. Guibas, P. Bailis, T. Hashimoto, and M. Zaharia, "Task-agnostic indexes for deep learning-based queries over unstructured data," *CoRR*, vol. abs/2009.04540, 2020. arXiv: 2009.04540.

[7] T.-Y. Lin *et al.*, "Microsoft coco: Common objects in context," in *Computer Vision – ECCV 2014*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds., Cham: Springer International Publishing, 2014, pp. 740–755, ISBN: 978-3-319-10602-1.

[8] L. Wen *et al.*, "DETRAC: A new benchmark and protocol for multi-object tracking," *CoRR*, vol. abs/1511.04136, 2015. arXiv: 1511.04136.