A Study on the Robustness of Object Detectors in Aqua-Farming

Rajarshi Biswas, Om Khairate, Mohamed Salman and Dirk Werth
August-Wilhelm Scheer Institute, Uni-Campus D 5 1, 66123 Saarbrücken, Germany
fi

Keywords: Deep Learning, Computer Vision, Industrial Application.

Abstract: In this paper, we study the robustness of state-of-the-art object detectors under transfer learning to detect live fishes swimming inside a fish tank. To overcome data limitations, we perform experiments in which we train these detectors with small amounts of annotated data and observe their robustness on out-of-domain data while tracking performance on in-domain test data. We compare YOLOv8l, RTMDet, RT-DETR, SSD-MobileNet and Faster-RCNN for performing dense object detection on images of fish schools obtained from an aqua-farm and observe their robustness on out-of-domain data from the MS COCO, ImageNet, and Pascal VOC datasets respectively. On the in-domain test set, we achieved the highest detection accuracy of 0.896 mAP with bounding boxes and 0.9214 mAP with instance masks using the YOLOv8l model. However, the same model exhibits a false positive rate of 55.77% on out-of-domain data from the MS COCO dataset. To mitigate false positive prediction we studied two different strategies, (1) re-training the models incorporating out-of-domain data and (2) re-training models by updating only the biases. We found that incorporating out-of-domain data to train the models leads to the highest reduction in false positive detection, however, this does not guarantee steady and high performance on the in-domain test data.

1 INTRODUCTION

In the field of computer vision, the problem of object detection is one of the fundamental challenges. The human visual mechanism can easily distinguish between foreground and background objects while simultaneously learning the underlying semantics of the image. However, these tasks still represent significant hurdles for a computer vision system. Developments in the past decade in neural network architectures, general-purpose GPU computing power, data availability, and storage options have transformed object detection. Deep convolutional neural networks proposed by Krizhevsky et al. (Krizhevsky et al., 2012) mark an important transition from handcrafted features in object detection. Subsequently, deep networks not only outperform traditional methods, but they also improve their performance with every iteration. The developments in object detection consistently focus on high accuracy and efficiency on standard datasets. However, the performance of these detectors on non-curated real-world datasets or under deployment in real situations sharply deteriorates. This is observed in terms of the drop in accuracy and lack of robustness as the trained models often have very high false positive rates on out-of-domain data.

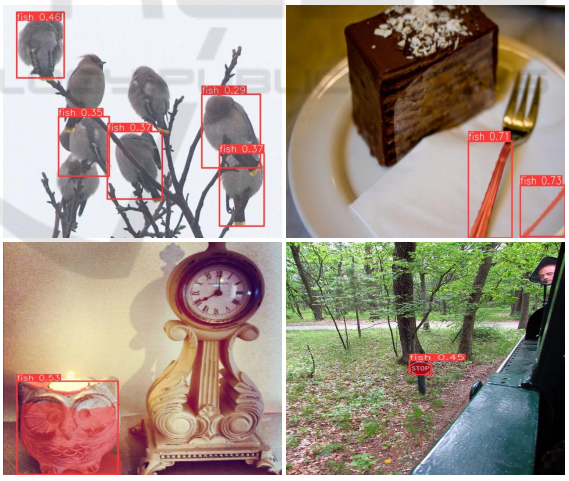

Figure 1: False positives of objects detected as fish on the MS COCO dataset.

In this paper, we study the robustness of five highly successful and state-of-the-art neural networks to perform object detection and instance segmentation. They are YOLO (Redmon et al., 2016), RTMDet (Lyu et al., 2022), RT-DETR (Zhao et al., 2024), SSD-Mobilenet (Howard et al., 2017), and Faster-RCNN (Ren et al., 2015) respectively on a real-world dataset for solving a practical business problem. The goal is

Biswas, R., Khairate, O., Salman, M. and Werth, D.
A Study on the Robustness of Object Detectors in Aqua-Farming.
DOI: 10.5220/0013342200003905
In *Proceedings of the 14th International Conference on Pattern Recognition Applications and Methods (ICPRAM 2025)*, pages 814-822
ISBN: 978-989-758-730-6; ISSN: 2184-4313

to perform dense detection of live fishes inside a tank for estimating their physical length and mass. Our dataset originates from an aquaculture farm in Germany and comprises images of fish schools of different species taken from an overhead camera looking down on the fish tank. This dataset is challenging compared to the standard datasets used in object detection. This is due to the fact that the dataset suffers from high intra-class variations resulting from pose, scale, occlusions, blur, etc. (refer to figure 2). This figure showcases a sample of instances that suffer from a high degree of occlusions, blur, pose and lighting artifacts.

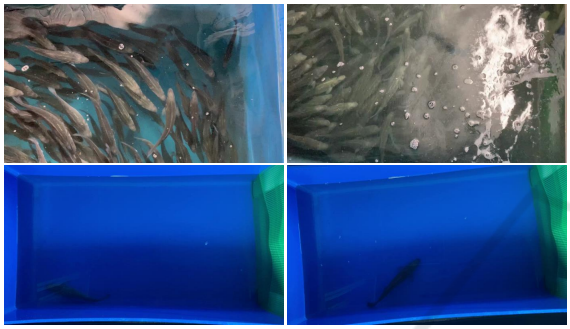

Figure 2: Challenging instances in our real-world fish dataset.

We employ the chosen neural networks to perform dense detection of fishes on images from this dataset. To train the models we use transfer learning and use different types of annotation strategy to create the training dataset. First, we create a dataset, with the least effort, comprising images containing a single annotated fish in each image, shown in figure 3. Following this, we create a second annotated dataset, with comparatively higher effort, comprising dense annotations of fishes in a cohort environment as seen in figure 4. The dataset comprises both single and dense annotations of fishes in a fixed proportion of 10:1.

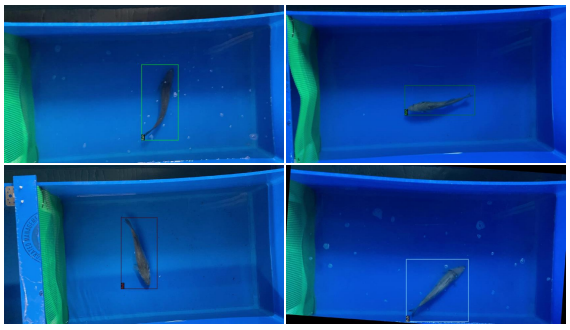

Figure 3: Bounding-box annotated images containing single fish.

More details on the datasets are provided in section 3. We extensively study the robustness of the chosen networks to perform dense object detection and instance segmentation. This is crucial to ensure stable and trustworthy performance under deployment in a business environment. To evaluate the robustness of the models we compute their false positive detection rates on out-of-domain data from three different sources. They are the ImageNet (Deng et al., 2009), the MS COCO (Lin et al., 2014) and the Pascal VOC (Everingham et al., 2010) datasets respectively.

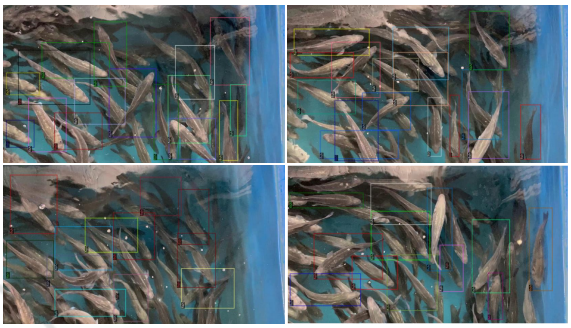

Figure 4: Densely annotated fishes with bounding boxes.

We organize our study in different experiments. In experiment (*E1*) we incrementally increase the percentage of out-of-domain data in the training set to re-train the models and monitor its influence on the false positive rate and detection accuracy. In experiment (*E2*), we perform instance segmentation using the best-performing neural network from *E1*. We also analyze the segmentation performance in terms of accuracy and false positive mask detection by gradually varying the amount of out-of-domain data used to re-train the model. Finally, in experiment (*E3*) we freeze all the weights in the chosen neural networks and train only the biases in the networks to study the effect it has on the false positive rate and detection accuracy. All the experiments are explained in detail under section 4. To summarize, our contributions in this work are the following:

- We analyze the robustness of five state-of-the-art neural networks YOLOv8l, RTMDet, RT-DETR, SSD-Mobilenet and Faster-RCNN for object detection on our challenging real-world dataset by monitoring their false positive rates on out-of-domain data from three different datasets.
- We analyze the robustness of instance segmentation performed by the best network among the five studied above for object detection.
- We analyze the impact of re-training only the biases of a network under transfer learning on de-

tection accuracy and robustness to save computational budget.

2 RELATED WORK

In the past two decades, object detection has captured the widespread interest of the vision research community. Consequently, many techniques have been developed that are now generating interest from various industries, such as autonomous driving, automated manufacturing, medical diagnosis and others. Object detectors can be broadly grouped into traditional approaches that employ hand-crafted features and deep learning based methods. The most well-known and best-performing traditional object detectors are the Viola-Jones detector (Viola and Jones, 2001), Histogram Of Oriented Gradients (HOG) detector (Dalal and Triggs, 2005) and Deformable Part Models (Felzenszwalb et al., 2008).

The introduction of deep convolutional neural networks (CNN) revolutionized the field of computer vision. Neural techniques such as AlexNet (Krizhevsky et al., 2012), VGG network (Simonyan and Zisserman, 2014), GoogleNet (Szegedy et al., 2015) and ResNet (He et al., 2016) were developed in quick succession and posted superior performance compared to traditional approaches. Girshick et al. (Girshick et al., 2014) introduced CNNs for object detection and subsequently object detection using deep CNNs has developed at a brisk pace. Neural network-based object detection can be further grouped into **double stage** and **single stage** techniques. **Double or Two stage** detection follows a coarse to fine strategy. The first stage generates the object proposals while the second stage performs categorical classification with refinement of the proposal locations. The Region-based convolutional neural network (R-CNN) is one of the first works in this direction. It uses selective search (Van de Sande et al., 2011) to generate the object proposals in the first stage that are classified with class-specific linear SVMs using features extracted, from the object proposals, with pre-trained ImageNet CNNs. This method suffered from slow speed and redundant computations. Spatial pyramid pooling network (SPPnet) (He et al., 2015) was developed to address this issue which sped up the detection process roughly 20 times than the R-CNN through feature sharing and computing the features from an image only once. Both R-CNN and SPPnet use multi-stage pipelines for feature extraction, object classification and bounding box regression which affect accuracy and speed. Fast R-CNN (Girshick, 2015) introduced a multi-task loss to jointly train the classifier and bounding box regressor. It also uses hierarchical sampling to enable feature sharing among proposals and efficient training. However, it still uses selective search to generate object proposals. The technique Faster-RCNN (Ren et al., 2015) employs a Region Proposal Network (RPN) to generate finer object proposals using CNN features. RPN creates object proposals in a sliding window manner upon which the added convolutional layers simultaneously classify the proposals as object/non-object and regress their locations. Faster R-CNN is widely preferred for detection tasks due to its efficiency and accuracy. Based on this architecture, other techniques (Lin et al., 2017a; Shrivastava et al., 2016) are developed to incorporate information in scales.

Single stage object detectors are simpler in architecture and offer faster detection speed. They can be compared to the RPN that also simultaneously performs object category classification and localization. The first single stage detector is the YOLO architecture (Redmon et al., 2016) proposed by Redmon et al. It takes the whole image as input, splits it into a grid, and then computes a fixed number of bounding boxes for each grid cell. Following this the probability of predicting an object in these bounding boxes are computed along with bounding box regression for localization. Yolo offers processing speeds from 45 frames per second (fps) to 155 fps in its fast version. Single Shot Multibox Detector (SSD) (Liu et al., 2016) proposed by Liu et al. performs predictions at different scales using feature maps from different layers of the network. Multiple scales approach boosts detection accuracy while handling of object classification and localization through convolutional layers allows integration of features at multiple scales. Lin et al. proposed focal loss (Lin et al., 2017b) to handle the training problems with foreground and background proposals in SSD. This helps SSD improve its performance.

However, a successful transition from effective academic techniques to full-fledged industrial applications requires that the techniques are robust and reliable under all possible circumstances. Some of the obstacles that affect the robustness of object detectors, trained under transfer learning, leading to poor performance under real-world situations are false positives, domain discrepancy and crafted adversarial examples. False positives are the predictions that do not match the ground truth annotations, are misclassified, or are poorly localized objects. Robustness to false positives is important to ensure reliable detection performance. So, in this work, we focus our attention on thoroughly studying the false positive detection char-

acteristics of five state-of-the-art neural networks to perform reliable object detection in an industrial setting using transfer learning.

3 DATASET

Our data comprises image frames of fish in isolated and group environments from 138 videos from an aquaculture farm in Germany. To promote quick adoption and ease of use, all videos are captured using a regular inexpensive camera positioned above the fish-tanks located on the premises of the aquaculture farm. The entire dataset is divided into two sub-datasets, that is, (i) a dataset containing mask annotation to perform instance segmentation of fishes (Segmentation dataset) and, (ii) a dataset with bounding box annotation to detect fishes (Detection dataset). This dataset contains images of fishes in cohorts as well as in isolation, i.e., images with a single fish in them.

The **Segmentation dataset** consists of a total of 2959 frames partitioned into train, validation and test sets respectively. Additional images, 1086 to be exact, from public datasets (Garcia-d'Urso et al., 2022; Saleh et al., 2020) are added to aid the generalization of the models. The **Detection** dataset comprises 2345 image frames extracted from videos containing a single isolated fish (refer to figure 3), along with 200 images of fish in cohorts. They are densely annotated using bounding boxes (refer to figure 4). The number of densely annotated instances is lower since it is extremely challenging to identify complete, non-overlapping fishes in the dense setting. We have to make a conscious decision during annotation on the extent of a fish that should be visible to be counted as a valid detection. This greatly increases the difficulty, cost and time required to produce the annotations.

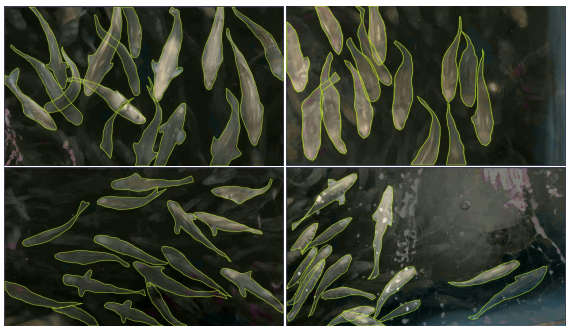

Figure 5: Annotations for instance segmentation in dense setting.

4 METHODOLOGY

In this work, we extensively study the robustness of five neural architectures, YOLOv8l, RTMDet, RT-DETR, SSD-MobileNet and Faster R-CNN respectively, for object detection and instance segmentation tasks through various experiments. All chosen models are trained using transfer learning on images obtained from an aquaculture farm to solve the specific tasks mentioned. Robustness to perform object detection is tested with all five neural architectures. Subsequently, we choose the best-performing network from this exercise, that is, YOLOv8l and test its robustness to perform instance segmentation. We study the robustness of the chosen architectures by primarily focusing on their false positive decision characteristics on out-of-domain data. This is because we use transfer learning to re-train the pre-trained networks on the task-specific dataset. We observed that under this training paradigm, the erroneous predictions are overwhelmingly false positives with almost insignificant to no false negatives. To represent out-of-domain data we use the images from ImageNet, MS COCO and PASCAL VOC datasets respectively.

In experiment (*E1*), we train the five chosen networks on the Detection dataset for object detection and observe their false positive rates on the three datasets mentioned earlier, representing out-of-domain data. We gradually incorporate out-of-domain data, in fixed proportions, into the training process of the models to study its influence on their false positive rates.

Under experiment (*E2*), we choose the model with the best performance from *E1*, that is, YOLOv8l, and train it on the Segmentation dataset to study the robustness of instance segmentation performed by it. The model is trained by gradually incorporating out-of-domain data in fixed amounts in the training set and analyzing its impact on robustness. In this way, we compare the instance segmentation and object detection performances.

Finally, in experiment (*E3*), we freeze all the weights of the networks apart from the biases in the layers of the network architecture. We train this setup for object detection using the Detection dataset. Subsequently, we observe the impact of training only the biases in a network on the false positive rate on out-of-domain data. For all the experiments described above, we also carefully observe detection accuracy on the in-domain test set through metrics, such as precision, recall and mean average precision.

5 RESULTS AND DISCUSSION

We trained the YOLOv8l, RTMDet, RT-DETR, SSD-MobileNet and Faster R-CNN networks for object detection to check their performance and the extent to which the task suffers from false positive predictions. For this purpose, we used the Detection dataset and follow the procedure in experiment *E1*. We gradually increased the proportion of out-of-domain data to train the models and observed its influence on the false positive rate across different out-of-domain datasets, i.e., ImageNet, MS COCO and Pascal VOC. Figure 6 depicts the percentage false positive rate characteristics of the chosen networks on the out-of-domain datasets. We observe that YOLOv8l exhibits the sharpest decline in false positive rates when out-of-domain data is incorporated during the training process (refer to figure 6).

From Figure 7, we can observe that all the networks considered in this study are adequately trained. Additionally, during the training process, we ensured that all networks neither overfit nor diverged. We also observed that under experiment (*E1*), all five networks exhibited the sharpest decline in false positive rates on the Pascal VOC dataset followed by the MS COCO and ImageNet datasets. We hypothesize that this can be due to the relatively simpler underlying distribution of the Pascal VOC dataset compared to MS COCO and ImageNet datasets.

The experiments that were conducted with YOLOv8l, RTMDet, RT-DETR, SSD-MobileNet and Faster R-CNN show that false positives are a consistent problem. For Faster R-CNN, the Region Proposal Network (RPN) is tasked with generating region proposals that are likely to contain objects. However, it works with convolutional feature maps, where background textures may cause false positives to be produced by the RPN. It may suggest regions containing background features that resemble things, such as textured areas like leaves, water and rocks, and incorrectly assume that these features could represent foreground items.

The issue arises due to the Region Proposal Network's (RPN) challenges in fully differentiating between actual objects and background patterns, especially in the early stages of detection. After the RPN suggests potential regions, Faster R-CNN extracts fixed-size feature maps from each proposal using ROI Pooling for classification. When these proposals contain a significant background, such as in cluttered environments, the model can mistakenly label the background as an object. Moreover, the pooling operation compresses spatial data, potentially losing fine details that are crucial to distinguish objects from the background. This can result in the model incorrectly identifying background areas as objects, Hence, producing false positives. Faster R-CNN uses predefined anchor boxes of varying scales and aspect ratios to propose regions. While these anchors are meant to correspond with different object sizes and shapes, they may not always perfectly align with the objects in the image. If an anchor box partially overlaps a background feature or contains both object and background elements, the RPN might generate a proposal for that area, contributing to false positives. This problem becomes more pronounced when background features have shapes or textures that resemble the objects the model is trained to detect.

Similar to the Faster R-CNN's use of a feature pyramid, SSD makes predictions directly from feature maps at multiple scales, but SSD predicts objects densely across multiple grid cells. Each grid cell is responsible for detecting objects within its receptive field. When background elements exhibit structured patterns or textures, e.g., waves, leaves, or shadows, the model can mistake them for objects. This is because SSD lacks explicit mechanisms to differentiate fine-grained details across spatial regions. It can overfit to background noise in highly textured scenes. MobileNet, as the backbone of SSD, is highly efficient and lightweight but focuses highly on reducing model complexity. This results in shallower feature maps that might not capture enough information to distinguish between subtle differences in background textures and objects. Due to this, the model may become overly sensitive to background patterns that resemble objects, leading to false positives, particularly in cluttered environments. SSD-MobileNet processes local image patches for object detection and does not have a strong mechanism to integrate global context (i.e., relationships between different regions of the image). In scenes with ambiguous background features, the model may not have enough contextual information to determine whether a detected region is part of the background or a real object. Without this global context, SSD can struggle to suppress background activations and might produce more false positives in complex or cluttered environments.

Similarly, RTMDet, a YOLO-based architecture, heavily relies on dense convolutional features extracted across various levels of a feature pyramid network (FPN). This design allows the model to detect objects at multiple scales, making it highly efficient in real-time scenarios. However, its reliance on dense feature extraction across the entire image grid can lead to complications, particularly when dealing

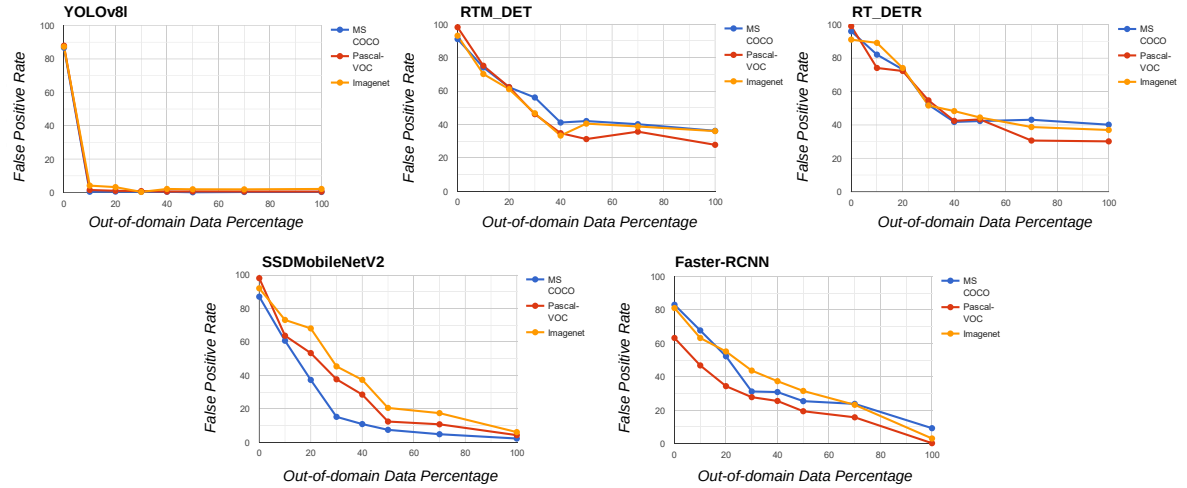

Figure 6: False positive rates for chosen networks under experiment E1.

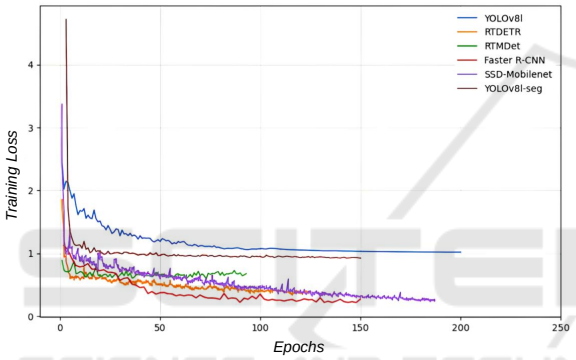

Figure 7: Training loss for networks under experiment E1.

with cluttered backgrounds or textures that resemble objects. Being an anchor-free detector, RTMDet predicts objects by focusing on key-point detection, specifically the center points of objects. In images with cluttered or complex backgrounds, the model may struggle to accurately localize the center of an object because multiple regions could present similar characteristics. Since RTMDet does not use predefined anchor boxes to guide the localization process (as anchor-based methods do), it can sometimes be influenced by background elements that mimic object shapes or textures, especially when those elements appear near the expected center points of the target objects. This can cause the model to generate false positives by assigning object labels to background regions that exhibit center-like features. RTMDet uses multi-scale feature maps to detect objects of various sizes, but the aggregation of these features from different layers can introduce noise when the background exhibits high variance across scales. For example, a large object (like a tree in the background) may have high feature activations at lower pyramid levels (which detect large objects), while finer textures (like leaves) might generate activations in higher pyramid levels (which detect smaller objects). This can cause feature interference where the model aggregates irrelevant background features from different scales, leading to over-sensitivity to background elements that resemble objects, resulting in false positives.

In a transformer-based system, like RT-DETR, the self-attention mechanism allows each pixel to potentially attend to every other pixel in the image. While this is a useful technique for modeling long-range dependencies and relationships, it can also cause the model to focus on background areas or other irrelevant areas, particularly in cases when the object's features visually resemble or overlap with those of the background. If the model is unable to distinguish between foreground objects and background regions, attention heads may erroneously recognise background attributes and link them with object classes when conducting object recognition tasks. False positives may emerge from the model misinterpreting background elements as foreground objects, particularly in circumstances when backdrop textures or patterns have properties similar to those of the target objects. Transformers primarily depend on learned connections, between parts of an image as opposed to neural networks (CNNs) or traditional object detectors that focus on specific areas and defined anchor boxes, e.g., in anchor-based methods. Since RT-DETRs are based on transformers and lack the biases found in CNNs that limit attention to regions of the image, they may erroneously link background details throughout the whole image with object detection. In detectors like RT-DETR that rely heavily on transformers to identify

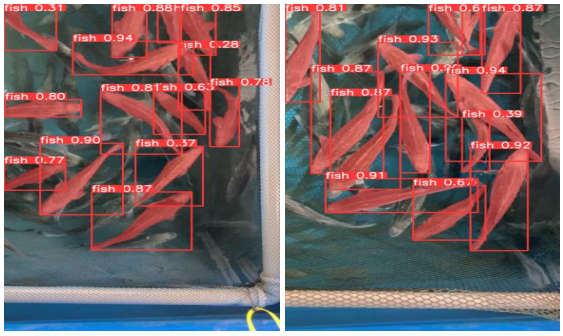

Figure 8: Segmentations performed by YOLOv8l on in-domain data.

objects in images and produce detection boxes, the choice of object queries is crucial for results. If these queries are not initialized correctly they may end up focusing on areas without objects which can lead to predictions and mistaken identifications of objects. Transformers can be affected by forms of interference in images like occlusions, compression artifacts and background noise such as reflections or shadows. The RT-DETR model may produce results due to its focus on connections. This could lead to even minor background disturbances being interpreted as important features. Transformers have the potential to amplify the influence of areas compared to CNN models which tend to be more robust due to their pooling mechanisms. Positional encodings, in transformers, are used to retain information. They are learned and not directly linked to the physical arrangement of objects. The global attention mechanism may be unable to distinguish between background regions and object boundaries if the model's learned positional encodings are insufficient or sub-optimal. This could result in false positives, where background areas are mistakenly regarded as objects.

We selected YOLOv8l from *E1* as it performed best among all the networks for experiment *E2*. In this experiment, we trained the YOLOv8l model on the Segmentation dataset (refer to section 3) for instance segmentation and obtained an impressive performance of 0.9214 mAP on the in-domain test data. Figure 8 shows examples of mask-based detection from the model on our in-domain test set. Subsequently, we checked the robustness of the model on out-of-domain data from the MS COCO, Pascal VOC and ImageNet datasets. We found that the model performed poorly on out-of-domain data from all three datasets with false positive rates of 55.77%, 47.08% and 59.08% on MS COCO, Pascal VOC and ImageNet respectively. Figure 1 shows some false positive examples on the MS COCO data. False positive detections on ImageNet and Pascal VOC datasets are

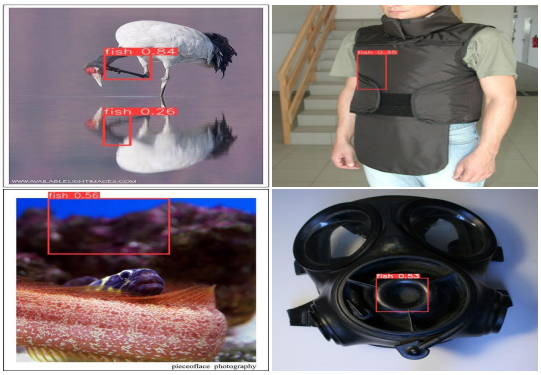

Figure 9: False positive predictions on ImageNet dataset.

Figure 10: False positive predictions on the Pascal VOC dataset.

depicted in figure 9 and figure 10 respectively.

Following the strategy explained in experiment (*E1*) we gradually incorporate out-of-domain data in our training set and observe the influence it has on the false positive rate. We re-train the model on all three datasets using a batch size of 16 and a learning rate of 0.01 with early stopping at 124 epochs. From

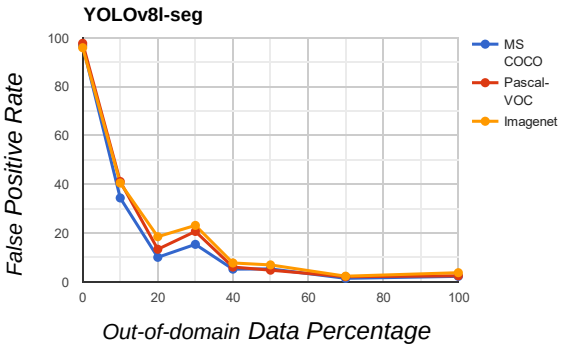

Figure 11: False positive rate of YOLOv8l for segmentation under experiment E2.

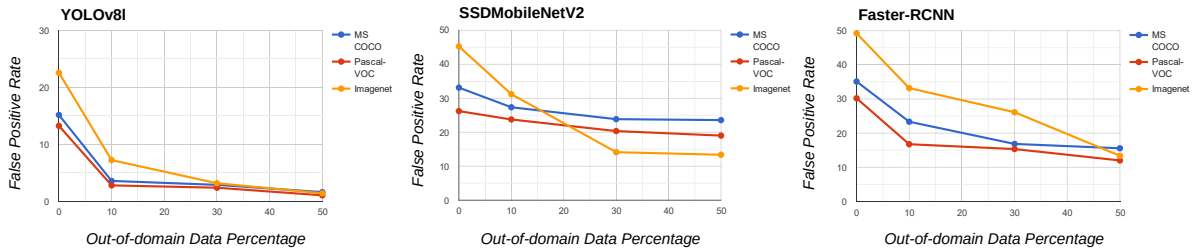

Figure 12: False positive rates for chosen networks under experiment E3.

figure 11 we can see that the false positive rate declines sharply but it also affects the accuracy of the model. On the in-domain test set the mAP drops to 41% with out-of-domain data at 50 percent level of the in-domain training data. It also exhibits shape hallucinations in certain images. One possible cause for this could be an imbalance in the losses. Bounding box loss, objectness loss, class loss, and mask loss are the several loss components that YOLOv8l segmentation model integrates into a single weighted total loss. A weighting error could cause some parts of the model (like the mask prediction) to become unduly dominant. For example, if the mask loss weight is disproportionately high compared to the weights for box loss or objectness loss, the model may prioritise mask formation even in situations where the bounding box detection is erroneous. This could also be the reason for the lower adaptation of the false positive rate to the usual YOLOv8l models.

In experiment (*E3*), we freeze all the weights apart from the biases in the layers of the chosen networks and train them with the in-domain dataset. Figure 12 depicts the results obtained from this experiment. We observe that YOLOv8l has the best performance against false positives when trained with different datasets. Also, we observe that under this experiment, the three networks achieve lower false positive rates with out-of-domain data from a larger dataset like ImageNet which can have a much more complex data distribution. This is in contrast with the results obtained under experiment (*E1*) where all three networks achieved better false positive rates on out-of-domain data.

6 CONCLUSION

In this work, we highlighted the problems associated with the robustness of state-of-the-art neural networks to perform reliable object detection in an industrial setting under transfer learning. We thoroughly studied the ability of five networks, YOLOv8l, RTMDet, RT-DETR, SSD-MobileNet and Faster R-CNN respectively, to handle out-of-domain data by carefully observing their false positive detection rates. To study this, we performed three different experiments *E1*, *E2* and *E3* with out-of-domain data from three different sources, that is, the MS COCO, the Pascal VOC and the ImageNet datasets. We found that under challenging industrial settings with limited availability of data, the neural network models are capable of achieving good performance on the in-domain data, but their performance sharply degrades on out-of-domain data. This is reflected by the high false positive detection rates from all the chosen networks on out-of-domain data from different sources. We found that the introduction of out-of-domain data in the training process of the models helps to lower the false positive detection rate, but does not completely solve the problem to the extent that the models can be deployed reliably. Interestingly, we observed that performing only bias updates during re-training of the models with out-of-domain data leads to the sharpest decline in false positive rates for all the networks, even on the largest out-of-domain dataset, i.e., ImageNet in our study. We intend to study this aspect in more detail in our future work.

ACKNOWLEDGEMENTS

This research was funded in part by the German Federal Ministry of Education and Research (BMBF) under the project FishAI (Grant number 031B1252B).

REFERENCES

Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, pages 886–893. Ieee.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee.

Everingham, M., Van Gool, L., Williams, C. K., Winn, J., and Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88:303–338.

Felzenszwalb, P., McAllester, D., and Ramanan, D. (2008). A discriminatively trained, multiscale, deformable part model. In *2008 IEEE conference on computer vision and pattern recognition*, pages 1–8. Ieee.

Garcia-d'Urso, N., Galan-Cuenca, A., Pérez-Sánchez, P., Climent-Pérez, P., Fuster-Guillo, A., Azorin-Lopez, J., Saval-Calvo, M., Guillén-Nieto, J. E., and Soler-Capdepón, G. (2022). The deepfish computer vision dataset for fish instance segmentation, classification, and size estimation. *Scientific Data*, 9(1):287.

Girshick, R. (2015). Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448.

Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587.

He, K., Zhang, X., Ren, S., and Sun, J. (2015). Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 37(9):1904–1916.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., and Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25.

Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. (2017a). Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2017b). Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., and Berg, A. C. (2016). Ssd: Single shot multibox detector. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pages 21–37. Springer.

Lyu, C., Zhang, W., Huang, H., Zhou, Y., Wang, Y., Liu, Y., Zhang, S., and Chen, K. (2022). Rtmdet: An empirical study of designing real-time object detectors. *arXiv preprint arXiv:2212.07784*.

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788.

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

Saleh, A., Laradji, I. H., Konovalov, D. A., Bradley, M., Vazquez, D., and Sheaves, M. (2020). A realistic fish-habitat dataset to evaluate algorithms for underwater visual analysis. *Scientific Reports*, 10(1):14671.

Shrivastava, A., Gupta, A., and Girshick, R. (2016). Training region-based object detectors with online hard example mining. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 761–769.

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9.

Van de Sande, K. E., Uijlings, J. R., Gevers, T., and Smeulders, A. W. (2011). Segmentation as selective search for object recognition. In *2011 international conference on computer vision*, pages 1879–1886. IEEE.

Viola, P. and Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001*, volume 1, pages I–I. Ieee.

Zhao, Y., Lv, W., Xu, S., Wei, J., Wang, G., Dang, Q., Liu, Y., and Chen, J. (2024). Detrs beat yolos on real-time object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16965–16974.