A Comparative Study on Convolution and Transformer based SOTA Object Detection paradigms and models

Parthiv A Dholaria
2021078
parthiv21078@iiitd.ac.in

Utsav Garg
2021108
utsav21108@iiitd.ac.in

Abhay Chowdhry
2021508
abhay21508@iiitd.ac.in

Abstract

This report investigates the performance of two state-of-the-art object detection paradigms: DETR (transformer-based) [Car+20] and EfficientDet [Sev20] (convolutional neural network-based). We achieve faithful reproduction of the results reported in the DETR and EfficientDet papers on the COCO [Lin+15] dataset. Subsequently, we evaluate the generalizability of these pre-trained models on the unseen Pascal VOC [Eve+12] dataset. This analysis unveils intriguing strengths and weaknesses in each approach. DETR demonstrates a surprising ability to detect small objects and leverage contextual information, while EfficientDet exhibits a more conservative approach with tighter confidence scores. The choice between these models depends on the specific application's needs, with factors like object size, context, and prioritization of precision versus recall playing a significant role. This work paves the way for future exploration of combining the strengths of both architectures for even more robust object detection.

1. Problem

Object detection is a fundamental computer vision task that involves identifying and localizing objects within an image. Traditionally, convolutional neural networks (CNNs) have been the dominant architecture for this task, achieving high accuracy on benchmark datasets. However, CNN-based approaches often require complex architectures and can struggle with certain challenges, such as detecting small objects or objects with complex relationships.

Recent advancements in transformer architectures, which have revolutionized natural language processing, have sparked interest in their application to object detection. This approach offers new possibilities for tackling the limitations of CNN-based methods.

This report aims to compare two state-of-the-art (SOTA) object detection paradigms: Convolutional Neural Networks (CNNs) exemplified by EfficientDet, and Transformers exemplified by DETR (DEtection TRansformer). We will delve into their architectures, strengths, and weaknesses to understand how they approach the object detection challenge and their relative effectiveness.

2. Model Details

This comparative study focuses on two specific object detection paradigms: DETR (representing transformer-based approaches) and EfficientDet (representing convolutional neural network (CNN)-based approaches). The selection of these models was deliberate and motivated by the following factors:

1. **DETR's Innovation**: While other transformer-based encoder-decoder architectures have been explored for object detection, some rely on pre-generated bounding box proposals. DETR, on the other hand, stands out as the first fully end-to-end approach. It eliminates the need for separate stages like region proposal networks, simplifying the overall architecture and potentially improving efficiency. This innovative aspect of DETR makes it a compelling choice for our comparison.
2. **EfficientDet's Generality**: While we considered YOLOv9 as a potential CNN-based counterpart, it focuses on a specific aspect of information flow within the network. We sought a more general-purpose CNN architecture for this comparison. EfficientDet fulfills this requirement. It leverages the well-established CNN approach while incorporating advancements like compound scaling and the BiFPN (Bi-directional Feature Pyramid Network) for improved feature fusion. This combination of established techniques with recent innovations makes EfficientDet a strong representative of the current state-of-the-art in CNN-based object detection.

By carefully selecting DETR and EfficientDet, we aim to provide a comprehensive comparison between the leading transformer-based and CNN-based object detection paradigms. Their contrasting architectures and strengths allow for a valuable exploration of the potential and limitations of each approach in this critical computer vision task.

Let us now shed some light into the inner works of DETR model and EfficientDet model.

2.1. DETR

DETR, introduced in the paper "End-to-End Object Detection with Transformers" by Nicolas Carion et al. (2020), represents a significant departure from traditional CNN-based object detection methods. It leverages the power of transformers, an architecture that revolutionized natural language processing, to directly address the object detection challenge.

1. **Transformer Encoder-Decoder Architecture**: Unlike CNNs which rely on multiple convolutional layers for feature extraction, DETR utilizes a standard transformer encoder-decoder architecture. The encoder processes the entire image through a CNN backbone, typically a ResNet variant. This backbone extracts high-level features that capture the spatial relationships and semantic information within the image. The decoder then attends to these encoded features, allowing it to focus on specific regions of interest.
2. **Object Queries and Set Prediction**: A key innovation in DETR is the use of a set of pre-defined object queries. These queries can be thought of as learnable templates that represent potential object detections. During training, the decoder interacts with the encoded image features and progressively refines these object queries, predicting bounding box coordinates and class probabilities for each potential object. This approach eliminates the need for separate region proposal and classification stages commonly found in CNN-based detectors.
3. **Hungarian Algorithm for Assignment**: Traditional object detection models often employ Non-Max Suppression (NMS) to remove redundant bounding boxes. DETR avoids NMS by employing the Hungarian algorithm for bipartite matching. This algorithm efficiently assigns predicted objects to ground truth annotations (objects labeled in the training data). It ensures that only the most confident prediction for each object remains, eliminating the need for post-processing steps like NMS.

2.2. EfficientDet

EfficientDet, introduced by Mingxing Tan et al. (2020) in the paper "EfficientDet: Scalable and Efficient Object Detection", is a high-performing object detection model based on convolutional neural networks (CNNs). It builds upon the success of EfficientNet, a family of CNN architectures designed for achieving high accuracy while maintaining computational efficiency.

1. **EfficientNet Backbone**: EfficientDet leverages the EfficientNet architecture as its backbone. EfficientNet utilizes a compound scaling method, which balances three key factors: network depth (number of convolutional layers), width (number of channels in each layer), and resolution (input image size). This method allows EfficientNet to achieve optimal performance under resource constraints, such as limited memory or processing power.
2. **Bi-directional Feature Pyramid Network (BiFPN)**: EfficientDet employs a BiFPN for feature fusion. Traditional object detection models often use a Feature Pyramid Network (FPN) to combine features from different network stages. However, FPN is limited in its ability to propagate high-resolution information from shallow layers to deeper layers. BiFPN addresses this limitation by allowing for bi-directional information flow between different network stages. This enables EfficientDet to capture objects at various scales more effectively, resulting in improved detection accuracy for both small and large objects.
3. **Compound Scaling:**: Similar to EfficientNet, EfficientDet utilizes a compound scaling approach to create a family of models with varying complexities. This family consists of several pre-defined EfficientDet models (e.g., EfficientDet-D0, EfficientDet-D7) that offer a trade-off between accuracy and computational efficiency. Users can choose the model that best suits their specific needs, depending on whether they prioritize higher accuracy or faster inference speed.

3. Result Replication

3.1. DETR

We successfully reproduced the exact results reported in the DETR paper "End-to-End Object Detection with Transformers" by Nicolas Carion et al. (2020). These results are presented in a LaTeX table below:

Table 1. Average Precision (AP) and Average Recall (AR) Metrics

Metric	IoU Threshold	Value
AP	IoU=0.50:0.95, area=all, maxDets=100	0.420
AP	IoU=0.50, area=all, maxDets=100	0.624
AP	IoU=0.75, area=all, maxDets=100	0.442
AP	IoU=0.50:0.95, area=small, maxDets=100	0.205
AP	IoU=0.50:0.95, area=medium, maxDets=100	0.458
AP	IoU=0.50:0.95, area=large, maxDets=100	0.611
AR	IoU=0.50:0.95, area=all, maxDets=1	0.333
AR	IoU=0.50:0.95, area=all, maxDets=10	0.533
AR	IoU=0.50:0.95, area=all, maxDets=100	0.574
AR	IoU=0.50:0.95, area=small, maxDets=100	0.312
AR	IoU=0.50:0.95, area=medium, maxDets=100	0.629
AR	IoU=0.50:0.95, area=large, maxDets=100	0.805

The reported values for loss functions, class error, and

cardinality error are presented both with and without scaling, as described in the DETR paper. Average Precision (AP) and Average Recall (AR) are provided for different areas (small, medium, large) and various maximum detection limits (maxDets).

3.2. EfficientDet

While the official EfficientDet implementation by Google Research is in TensorFlow, we utilized a PyTorch adaptation [Sev20] for this work. This adaptation, as documented on the GitHub repository, removes unnecessary biases in convolutional layers followed by batch normalization, resulting in a slight reduction in model parameters.

We achieved the following results using the PyTorch adaptation:

Table 2. Average Precision (AP) and Average Recall (AR) Metrics

Metric	IoU Threshold	Value
AP	IoU=0.50:0.95, area=all, maxDets=100	0.420
AP	IoU=0.50, area=all, maxDets=100	0.611
AP	IoU=0.75, area=all, maxDets=100	0.448
AP	IoU=0.50:0.95, area=small, maxDets=100	0.231
AP	IoU=0.50:0.95, area=medium, maxDets=100	0.475
AP	IoU=0.50:0.95, area=large, maxDets=100	0.582
AR	IoU=0.50:0.95, area=all, maxDets=1	0.338
AR	IoU=0.50:0.95, area=all, maxDets=10	0.531
AR	IoU=0.50:0.95, area=all, maxDets=100	0.563
AR	IoU=0.50:0.95, area=small, maxDets=100	0.341
AR	IoU=0.50:0.95, area=medium, maxDets=100	0.627
AR	IoU=0.50:0.95, area=large, maxDets=100	0.741

The results obtained for EfficientDet using the PyTorch adaptation are comparable to those reported in the original paper "EfficientDet: Scalable and Efficient Object Detection" by Mingxing Tan et al. (2020). However, we observed a slight decrease in Average Precision (AP) of approximately 1.5

4. Inference and Analysis

Having successfully reproduced the results for DETR and EfficientDet on the COCO dataset, we can now leverage these pre-trained models to gain further insights into their strengths and weaknesses. In this section, we will evaluate the performance of both models on the Pascal VOC dataset. This evaluation on an unseen dataset allows us to assess how well these models generalize to new data with the same object detection task. By analyzing their performance on Pascal VOC, we can identify potential biases or limitations inherent to each approach.

The Pascal VOC dataset presents a valuable opportunity to explore the following:

- Generalizability: How well do DETR and EfficientDet, trained on COCO, perform on a different dataset with similar object detection challenges?
- Strengths and Weaknesses: Does either model exhibit specific strengths or weaknesses when applied to the Pascal VOC dataset? Are there particular object categories or image characteristics that pose a challenge for either approach?

We calculated the IoU metrics for 20 images in the Pascal VOC dataset for the detection images by both models and results were as follows:

1. **DETR**: 0.87657
2. **EfficientDet**: 0.84210

Through this analysis, we aim to gain a more comprehensive understanding of the capabilities and limitations of DETR and EfficientDet for object detection tasks. We infer the prediction on a threshold of 0.3 for each

4.1. Small Object Detection

As we can see in Fig 1. that DETR is surprisingly able to identify small objects such as the watch which was classified as a 'clock'. The reason is surprising is because DETR paper specifically mentions that the models performs lower on small objects. However running them on an unseen image shows that DETR was able to find them better than the EfficientDet as shown in Fig 2. This is further seen by the detection of the partially visible people with a lower probability which EfficientDet was not able to recognize.

4.2. Context

Context is perhaps the most interesting aspect of this analysis. Fig 3 shows a cup being identified from where the map is playing a trumpet, while the same is not the case in EfficientDet. This is, of course an incorrect detection, but interesting nonetheless because a possible explanation for the same is the use of surrounding context in case of DETR, this motivates a possible reason that the way the man's mouth and hand are structured with the trumpet's opening align very often with how the model would have encountered a lot of "cups" hence learning not only the structure of the cup but also the surrounding content for a cup and hence classifying a similar overall structure as one (Notice the lower confidence in the cup in Fig 3.). Clearly there is no notion of attention and hence no cup in Fig 4.

4.3. Confidence Disparity

We can notice is all the image pairs shown that there is a clear confidence disparity between DETR and EfficientDet, which manifests itself as the follows:

- Identification of objects not present in the image in case of DETR since the cutoff is too low compared to the confidence range most objects are given

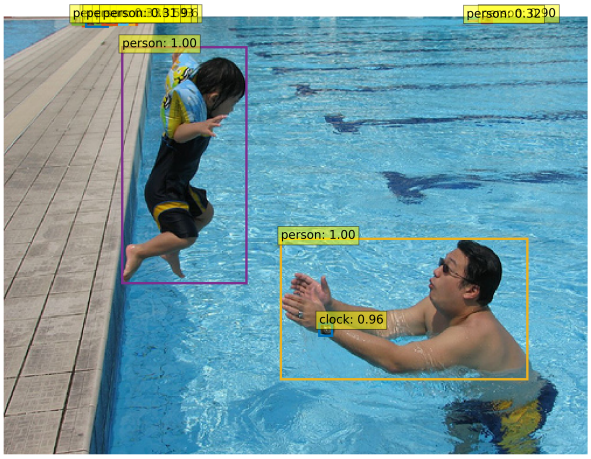

Figure 1. DETR Inference

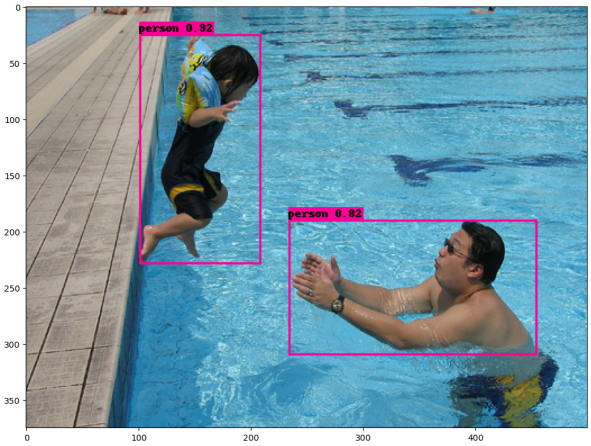

Figure 2. EfficientDet Inference

- Low confidence even on correct objects in case of EfficientDet often causes it to be not being able to identify objects because they couldnt cross the threshold. We can clearly see in a lot of the provided examples that the correct box barely crosses the threshold.

4.4. Camouflage ?

As we can see in Fig 3. that EfficientDet was able to identify the cup present at the bottom left of the images while DETR was not able to do the same. As one might notice, the cup is even hard to be noticed by a human in the first go because of its small size and the how its rather hidden, and yet the EfficientDet was able to identify it.

4.5. Generalizability

Fig 5. shows how clearly DETR better generalizes to unseen environments and unlikely setups with its precise identification of the plant in the building which a human eye would have missed as well, while the efficientDet is unable

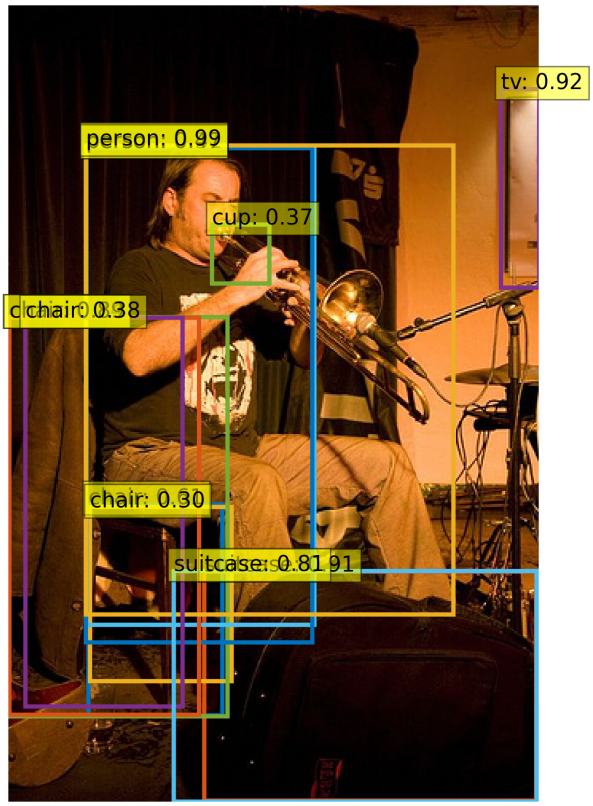

Figure 3. DETR Inference

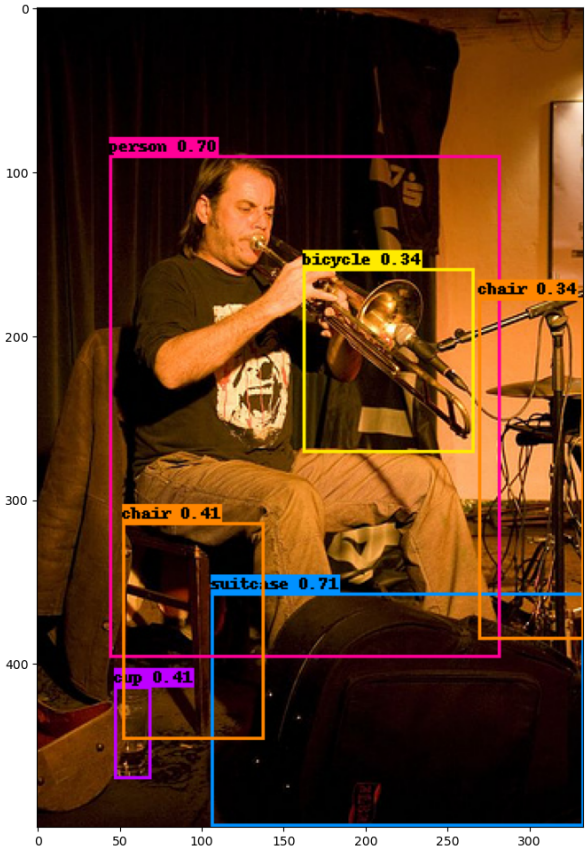

Figure 4. EfficientDet Inference

to do that.

5. TIDE Analysis

To evaluate the performance of object detection models, researchers often rely on metrics that consider both the accuracy of the detections and the efficiency of the model. TIDE (Task-oriented Inference for Dense object Detection) is a metric that combines these two aspects. In the context of DETR and EfficientDet models, TIDE can be used to assess how well these models balance accuracy and efficiency for object detection tasks.

5.1. DETR Results

Table 3. bbox AP @ [50-95] (Part 1)

bbox AP @ [50-95]					
Thresh	50	55	60	65	70
AP	7.38	7.36	7.30	7.30	7.30

Table 4. bbox AP @ [50-95] (Part 2)

Thresh	75	80	85	90	95
AP	7.30	7.16	6.77	6.19	4.13

Table 5. Main Errors

Main Errors						
Type	Cls	Loc	Both	Dupe	Bkg	Miss
dAP	0.00	0.89	0.82	0.06	1.78	0.20

Table 6. Special Error

Special Error		
Type	FalsePos	FalseNeg
dAP	2.13	0.34

5.2. EfficientDet Results

Table 7. bbox AP @ [50-95] (Part 1)

bbox AP @ [50-95]					
Thresh	50	55	60	65	70
AP	7.06	6.99	6.88	6.82	6.70

5.3. Analysis

- DETR: Achieves a higher peak AP of 7.38 compared to EfficientDet's 7.06. This indicates potentially better overall detection accuracy for DETR.

Figure 5. DETR Inference

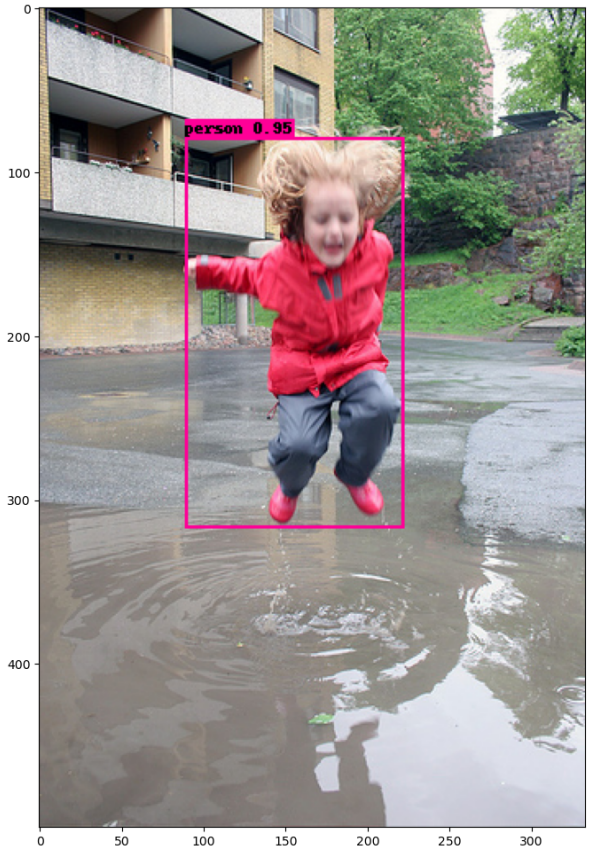

Figure 6. EfficientDet Inference

Table 8. bbox AP @ [50-95] (Part 2)

Thresh	75	80	85	90	95
AP	6.57	6.47	6.06	5.19	2.72

Table 9. Main Errors

Main Errors						
Type	Cls	Loc	Both	Dupe	Bkg	Miss
dAP	0.00	0.74	0.00	0.04	0.79	0.08

Table 10. Special Error

Special Error		
Type	FalsePos	FalseNeg
dAP	1.85	0.14

- EfficientDet: Maintains a more consistent AP across different IoU (Intersection over Union) thresholds compared to DETR, whose AP drops faster at higher thresholds.
- DETR: Has a higher number of false positives (2.13) compared to EfficientDet (1.85). This suggests DETR might make more detections overall, potentially impacting both accuracy and efficiency.
- Localization Errors (Loc): DETR shows a higher dAP (0.89) for localization errors compared to EfficientDet (0.74). This suggests DETR might struggle more with accurately placing bounding boxes around objects.
- Background Errors (Bkg): DETR has a significantly higher dAP (1.78) for background errors compared to EfficientDet (0.79). This indicates DETR might be more prone to misclassifying background pixels as objects.
- Missing detections (Miss): DETR has a higher dAP (0.20) for missing detections compared to EfficientDet (0.08). This suggests DETR might miss some objects entirely compared to EfficientDet.

Overall, while DETR shows a slightly higher peak AP, its higher false positives and background errors might inflate its overall detection count, impacting efficiency. EfficientDet, with its lower false positives and background errors, might be a better choice for tasks where precision is crucial.

6. Member's Discussions

6.1. Parthiv

We started this project with an aim to bring out an extensive comparative analysis of very basic CNN and Transformer based architectures and study how they perform in detecting objects efficiently, where they are lacking and what shall be the future directions to explore. Our study reveals very interesting findings. One of them that I found fascinating was the ability of Transformer based DETR's of generalizing and identifying the common patterns in human class. For example, misprediction of trumpet to cup. Although a misprediction it shows how well it has understood the surrounding context of alignment of humans with cups that are used for drinking commonly. The same behavior was seen for small objects which are very hard to see for a normal human eye and that too with a high confidence score of 0.72.This shows that the model has understood the common habits of humans very well with different instruments. Secondly, in a very challenging scenario which even the human eye cannot detect, Detr was able to detect a person hiding behind a person, indicating how advanced these Deep Learning models have become surpassing humans.Third, EfficientDet's ability to identify small scale object in a camouflage environment was really a good observation by the model.. Places that I feel that can be improved further is EfficientDet's ability to predict the bounding boxes with a higher confidence. Even for a single image with only a cat in it, EfficientDet was able to predict the bounding box but with very little confidence score on PascalVOC2012. For DETR, exploration on the side of generalizing better for the human class should be an interesting task to do.

6.2. Utsav

The mAP with IoU=0.50:0.95 shows comparable and good results for both EfficientDet and DETR on the COCO dataset. Although we can see from the inference results that for small and medium area objects, EfficientDet performs slightly better than DETR, while the result is opposite in case of large area objects. The above result is expected to happen given EfficientDet employs BiFPN in its architecture. Secondly, the DETR is able to give a better confidence score utilizing its context learning as well. We hypothesize this observation because the model detects a cup instead of trumpet or a watch on a person's hand or a person inside the train as that is what is generally present in scenarios. Although these predictions are incorrect at times and hence DETR requires high IoU thresholding in range of 0.9. Therefore in cases such as medical object detection or manufacturing defect detection it is suitable to use convolution based networks where we need less number of false positives.

6.3. Abhay

The analysis revealed some interesting findings, such as the DETR being able to recognize the plant and misclassified the cup. I find this to be rather fascinating how something originally created for language is being adapted to vision tasks, and even more surprising about how good it performs! Few of our findings definitely matched with the intuition on which these adaptations were made such as the effect of local context for attention actually affecting an object class, this shows how are models are evolving to think

in patterns which our mind uses as well, using the context of a mirror to distinguish between a real person and a reflection. These are however, of course, mere backward reasoned speculation, which at the same time also open endless possibilities for future exploration. The differences in generalizability by the plant and the cup are two prime differences that should be further explored in far greater detail, understanding their actual cause could reveal into the insights of the model's strengths and weaknesses.

7. Conclusion

This analysis of DETR and EfficientDet on the unseen Pascal VOC dataset provided valuable insights into their generalizability and performance characteristics. While both models achieved reasonable performance, their strengths and weaknesses became more apparent when applied to a new data distribution.

DETR displayed a surprising ability to detect small objects, outperforming EfficientDet in this regard. Furthermore, DETR appeared to leverage contextual information to make inferences, leading to some interesting, albeit incorrect, detections. However, a potential downside of DETR is the observed confidence disparity, where the chosen threshold might lead to the inclusion of false positives or the exclusion of correct detections with lower confidence scores.

EfficientDet, on the other hand, exhibited a more conservative approach, potentially leading to missed detections, particularly for objects with low visibility or small size. However, EfficientDet's confidence scores seemed to be more tightly clustered, potentially reducing the issue of false positives. Additionally, EfficientDet demonstrated a capability to identify well-camouflaged objects that might be challenging even for human observers.

In conclusion, both DETR and EfficientDet offer valuable capabilities for object detection tasks. DETR excels at identifying small objects and appears to leverage contextual information, while EfficientDet demonstrates a more conservative approach with tighter confidence scores. The choice between these models might depend on the specific requirements of the application, with factors such as object size, context, and prioritization of precision versus recall playing a significant role. Future work could involve exploring different confidence score thresholds and potentially combining the strengths of both architectures to achieve even more robust object detection.

References

[Eve+12] M. Everingham et al. *The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results*. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html. 2012.

[Lin+15] Tsung-Yi Lin et al. *Microsoft COCO: Common Objects in Context*. 2015. arXiv: 1405.0312 [cs.CV].

[Car+20] Nicolas Carion et al. *End-to-End Object Detection with Transformers*. 2020. arXiv: 2005.12872 [cs.CV].

[Sev20] Sevakon. *EfficientDet: PyTorch Implementation*. Accessed: May 11, 2024. 2020. URL: https://github.com/sevakon/efficientdet.

A. Further Comprisons

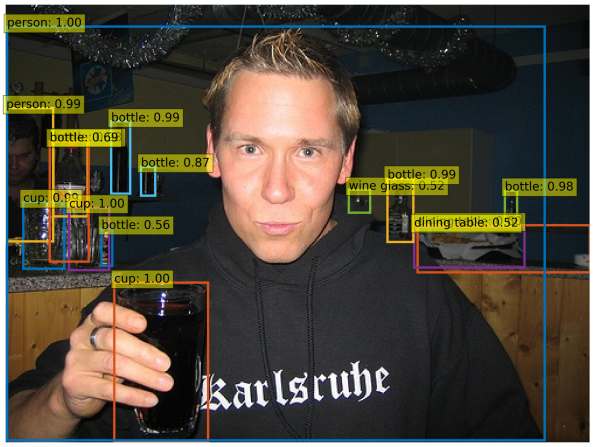

Figure 7. DETR Inference

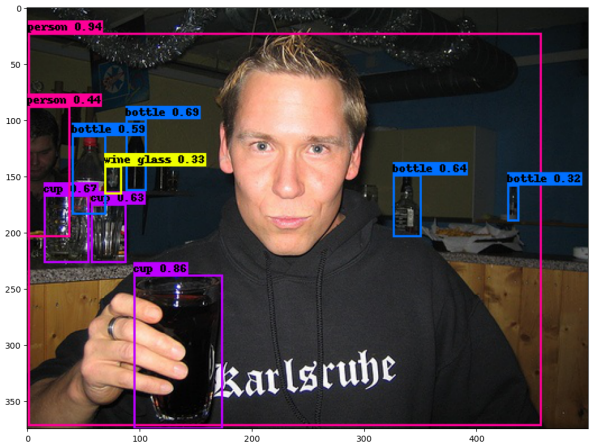

Figure 8. EfficientDet Inference

Figure 9. DETR Inference

Figure 10. EfficientDet Inference

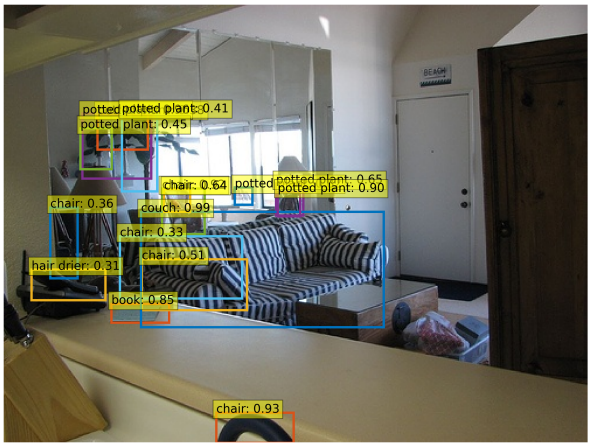

Figure 11. DETR Inference

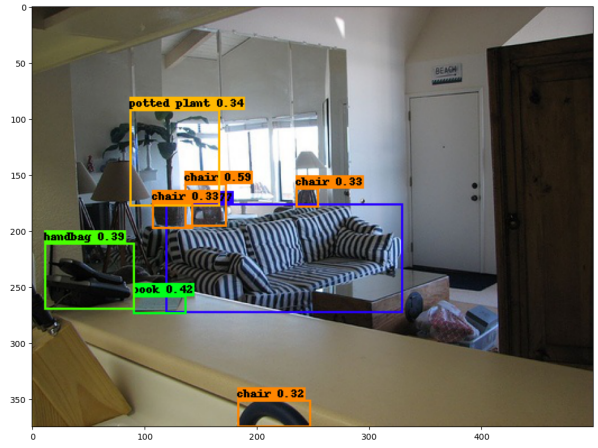

Figure 12. EfficientDet Inference

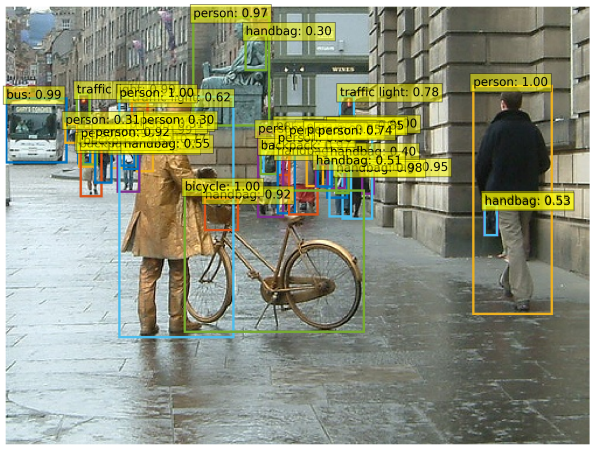

Figure 13. DETR Inference

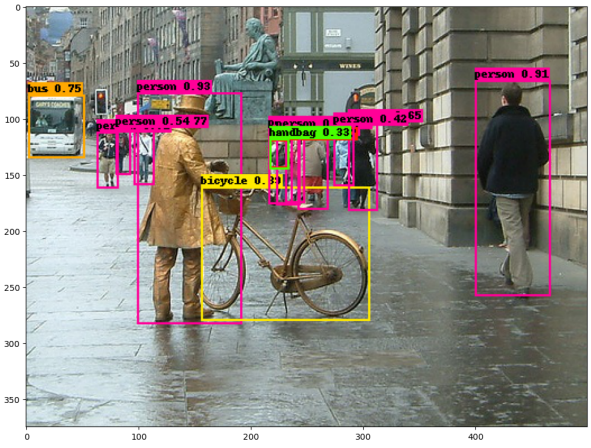

Figure 14. EfficientDet Inference

Figure 15. DETR Inference

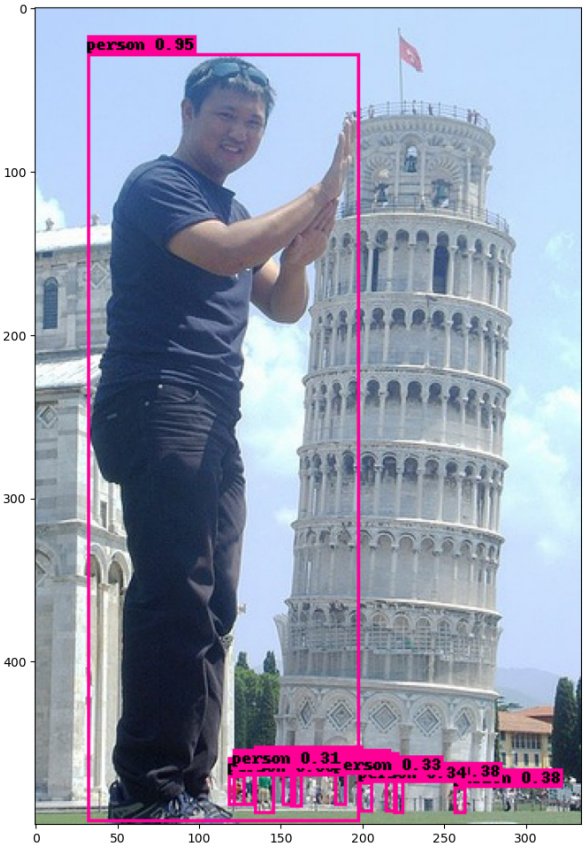

Figure 16. EfficientDet Inference

Figure 17. DETR Inference

Figure 18. EfficientDet Inference

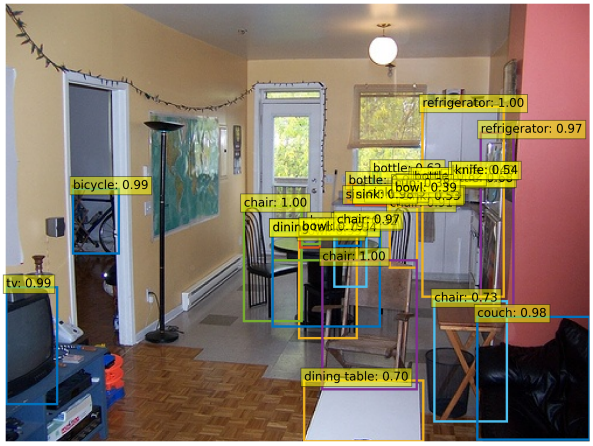

Figure 19. DETR Inference

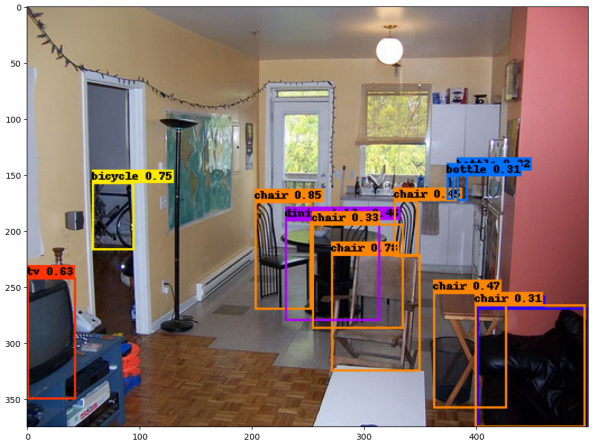

Figure 20. EfficientDet Inference