Digital Object Identifier 10.48165/iitmjit.2026.12.1.13

AI-Based Object Detection Architectures for Real-Time Precision Targeting Systems: A Comparative Analysis of CNN and Transformer Models

Keshav Tyagi[1], Priyanka Bhutani[2]
[1,2]Department of Computer Science Engineering, University School of Information, Communication and Technology
[1]ktyagi475@gmail.com, [2]priyanka.b@ipu.ac.in

Abstract - Object detection Artificial Intelligence Object detection is now a key concept in the contemporary precision targeting systems utilized in unmanned combat aerial vehicles, missile guidance units, and autonomous surveillance platforms. Although the convolutional neural network (CNN) detectors dominate the real time implementation, the transformer based ones have global context modelling which can contribute in improving the robustness of detection in the complex environments. Nonetheless, the accuracy versus computational efficiency trade-off when deployment limits are put in charge is under-developed. In this paper, we are going to provide a comparative analysis of a typical CNN-based detector (YOLOv5) and a transformer-based detector (DETR) on controlled runtime on the COCO 2017 validation set. Models were tested on a GPU-based system and were tested based on the accuracy of the detection, the inference latency, the number of parameters, and its applicability. It has been shown that YOLOv5 has a much higher real-time throughput and reduced memory overhead, whereas DETR has better localization consistency with more stringent IoU thresholds. The results indicate that there is a serious efficiency-context modelling trade-off of architectural selection in precise targeting systems. Despite the advantages of representation provided by transformer-based models, convolutional detectors have become more operational in operational scenarios that are latency-sensitive in defence tasks. Future research directions of the hybrid architectures and hardware-sensitive optimization are also given in the paper.

Keywords - Artificial Intelligence, Object Detection, Precision Targeting, CNN, Transformers, Real-Time Systems, Edge Computing, Adversarial Robustness.

1. Introduction

The calculation core of modern accuracy is the object detection with the help of Artificial Intelligence (AI) targeting systems: the use of targeting systems in unmanned combat aerial vehicles (UCAVs), autopilot surveillance. Smart fire-control systems, systems, and missile control teams. The significance of precision in detection is not the only issue with such mission-critical settings, the performance of operations. The inference latency, the computational footprint and there are also factors that affect in these settings, capability to endure under hostile conditions in the battle field of missions [8].

CNN-based object detectors have dominated real-time object detection in the last decade. Such architectures as YOLO [1], the SSD [2] and later optimization variants such as YOLOv4 [3] and YOLOv5 [4]) demonstrated acceptable speed accuracy trade-offs with single-stage detection pipelines. These models are high throughput as well and the fusion of feature extraction and bounding box regression into single forward passes, hence it is appropriate on embedded and edge based defence systems.

Nevertheless, convolutional detectors do to rely on localized receptive fields and thus only provide global information in an incrementally hierarchical manner by aggregation. Under cluttered working conditions with occlusion, camouflage and high densities of targets, limited long-range dependency modelling can be important in influencing the detection stability and contextual reasoning.

A different paradigm is presented by the transformer-based vision models, which use global self attention mechanisms. DETR [6] redefined object detection as a direct set prediction task, which did not involve the use of

hand-designed anchor designs and non-maximum suppression steps. Deformable DETR [7] enhanced the conversion efficiency with maintaining the global context modelling. Equally, Vision Transformers (ViT) models proved that patch-based tokenization can be based on multi-head attention instead of a traditional convolutional feature extractor [8]. Such models clearly represent space relationships throughout the scene, which may be very useful when dealing with complex battle scenarios.

Transformer based detectors are characterized by much greater computational overheads and training. Their quadratic attention scaling is more expensive in memory consumption and inference latency that can be limiting when deployed in edge-constrained precision targeting systems where deterministic response time and energy efficiency are of concern [10].

Despite the fact, that both CNN and transformer architecture have been widely tested on databases like COCO [14] and DOTA [15], systematic testing at defence like run-to-run demands has not been highly evaluated. Precisely, there is a lack of comparative evaluation of accuracy-latency trade-offs, hardware viability, and operation suitability in precision targeting tasks.

This paper fills this gap with the help of a two-phase model:

(i) Analytical comparison of representative CNN-based and transformer-based detection architectures using reported performance metrics, and

(ii) Experimental comparison of pre trained YOLOv5 [4] and DETR [6] models using controlled experimental conditions with the same runtime conditions.

Instead of concentrating on benchmark accuracy, the goal is to test the operational feasibility in real time precision targeting environment where reliability, computational efficiency and deployment feasibility are also mission critical.

2. Problem Statement

Although object detection with the help of deep learning has progressed quickly, there are concerns about deploying object detection models in precision targeting systems that go beyond benchmark accuracy. The defence environments present severe limitations on the inference latency, computational efficiency, sensitivity to environmental variability, and resistance to adversarial manipulation.

YOLO and SSD are CNN-based detectors that are highly frame rate and hardware efficient [1]– [4] and can thus utilize them in embedded applications. Their dependence on localized convolutional receptive fields, however, might be a weakness in contextual reasoning in dense or occluded battlefield situations. Transformer models like DETR and Deformable DETR [6], [7] can focus on the global dependency modeling by using self-attention, but they come with a large computational footprint and slower convergence characteristics, which can impose a challenge on the feasibility of deploying edges.

There are also other conditions that are likely to occur on the battlefield such as low lighting, atmospheric effects, motion blur, camouflage, and confounded targets. Empirical research points at degradation of performance of object recognition system in such unfavourable circumstances [8]. The effects of a false positive or lack of detection by the operational systems of precision targeting systems may be vital and require reliability that goes beyond the traditional benchmarks of the dataset.

There are also security vulnerabilities, which complicate the deployment. It is demonstrated that adversarial perturbations can be used to control object detection outputs, which is a cause of worry about the issue of robustness in military AI systems [11]. In addition, distributed edge intelligence is becoming more and more important in real-time target recognition, with computation and power limitations of the architectural choice directly influencing architecture choice [10].

Even though current literature has effectively analysed detection architectures using standardised datasets, like those of COCO [14] and DOTA [15], few studies analytically measure their applicability and suitability to defence-imposed runtime constraints. In particular, it is not clear enough how:

- Accuracy -latency tradeoff analysis in controlled hardware conditions.
- CNN versus transformer detector comparative evaluation in terms of edge deployment.
- Adversarial and environmentally degraded strengthening.

These aspects need to be addressed to evaluate that the architectural advances have been translated into operational effectiveness in the mission-based precision targeting systems.

3. Literature Review

Conventional Object detection Objects Deep learning Object detection has developed in two main architectural paradigms, convolutional and transformer based.

First single-stage CNN detectors including YOLO [1] and SSD [2] have shown that integrated detection networks can support real-time performance without region proposal networks. Following developments, such as YOLOv4 [3] and YOLOv5 [4], backbone efficiency, feature pyramids and loss strategies were optimized to improve the speed-accuracy ratio. The latest reviews of YOLOv8 to detect drones ensure that contemporary CNN variants possess a high real time performance of aerial surveillance tasks [9].

Although CNN architectures are based on hierarchical feature aggregation, they implicitly represent global context using deeper and deeper layers. This design has been successful with realtime systems and can be limited when dealing with complex scenes that have variation in size and heavy object distribution.

Detectors based on transformers also brought a structural change. DETR was reformulated as a global self-attention based direct set prediction making it direct to prediction, removing anchor engineering and non-maximum suppression heuristics (DETR) [6]. The deformable DETR [7] enhanced faster convergence and less redundancy in computation by sparse attention. Vision Transformer models [8] also proved that patch-based tokenization with multi-head attention could be used to generalize the learning of visual representations beyond convolutional interactions.

The AI-based object recognition systems have been tested in the conditions of defence use, which revealed that the response was sensitive to the lighting variation, occlusion, and atmospheric interference [8]. Edge AI recognition systems in UAVs have highlighted the need to have efficient operations with minimal latency in military systems that are distributed [10].

Security factors have become eminent too. Research on adversarial robustness demonstrates that object detectors are susceptible to attacks that either generate misidentification or manipulate bounding-boxes [11]. These findings demonstrate the necessity of reliability tests other than traditional accuracy measures.

Although a great progress has been achieved in the sphere of architecture, the literature that explores the subject of limitations of operational deployment in specific, does not provide any comparative analysis of the topic. Most of the literature boast of their ability over large-scale datasets such as the COCO [14] or aerial datasets such as DOTA [15] but few of them quantify the trade-offs between the complexity of their architecture, the inference and deployment determinism in precise target applications.

This break keeps inviting a methodological comparative research on the foundation of the practicality and not the excellence on the benchmark secluded.

4. Methodology

In this paper, a systematic comparative analysis will be conducted to determine convolutional based object detection models and transformer-based object detection models on real time objectives of precision limitations. Not only would the detection accuracy of the model be assessed, but the computational capabilities and deployability in edge-constrained environments would also be assessed.

4.1 Model Selection

To ensure the architectural diversity and relevance on operations, the number of two representative architectures was chosen:

- • **YOLOv5 [4]** an example of a single-stage CNN-based detector that is optimized with respect to inference in real-time and embedded implementation.

- **DETR (Detection Transformer)** [6] — is a transformer-based detector that uses global self attention and set-based prediction.

YOLOv5 is an advanced convolutional detection pipelines that focus on speed-accuracy balance and DETR is an end-to-end transformer detection that has explicit global dependency models. The choice is made to guarantee architectural contrast and still guarantee feasibility of controlled experimentation.

Standardized evaluation and the removal of training bias were ensured by using pre trained weights that were trained in the COCO dataset [14].

4.2 Dataset Configuration

Our experiments were conducted on COCO 2017 [14] validation set. We used only the selected subset of images to perform the runtime benchmarks to make the tests manageable and at the same time to have useful statistics.

COCO was selected on the basis of a number of reasons:

- It is the common yardstick applied by everyone.
- It supports multi-scale, multi-class evaluation.

- It facilitates a comparison between the numbers and literature.

Admittedly, COCO is not defense-specific but its combination of scales, densities and background complexity can still provide us with a sounding point in terms of comparing other architectures.

4.3 Experimental Environment

All of this was done on Google Colab with an NVIDIA T4 (16GB VRAM) with a CUDA-enabled run time.

Environment specifications:

- Python 3.x
- PyTorch framework
- Existing ready-made model applications.
- GPU-based inference

We ensured that the two models were given the same conditions so that the measurement of latency and throughput would be equal.

4.4 Evaluation Metrics

We computed the quality of detection as well as deployment energy hence we considered the measures of accuracy and efficiency.

Accuracy Metrics

- **mAP@0.5** (mean Average Precision at IoU threshold 0.5)
- **mAP@[0.5:0.95]** (COCO standard metric)
- **Intersection over Union (IoU)**

These metrics quantify detection precision and localization quality.

Efficiency Metrics

- Inference Latency (ms/image)
- Frames Per Second (FPS)
- Model Size (MB)
- Parameter Count (Millions)

The efficiency is important in that the response time and computing load may break or make the operation in the real time precision-targeting.

4.5 Comparative Framework

We made two comparisons, the first comparing accuracy and efficiency on the same hardware, and the second comparing the implementation:

1. **Quantitative Evaluation:**
   We made two comparisons, the first comparing accuracy and efficiency on the same hardware, and the second comparing the implementation.
2. **Operational Assessment:**
   Interpretation of results in the context of precision targeting constraints, including:
   a. Real-time responsiveness
   b. Edge deployment feasibility
   c. Computational scalability
   d. Reliability under constrained resources

5. Experimental Setup

In this section, we will discuss the practical pipeline that we have employed in order to test YOLOv5 and DETR in those controlled settings.

5.1 Implementation Framework

All the runs were performed on Colab on the GPU of PyTorch. The official pre trained models were used so that we could avoid additional variability and so that all can be reproducible.

- YOLOv5: Ultralytics official repository [4]
- DETR: Official Facebook AI Research implementation [6]

Our comparison was strictly architectural since the inference-time weights were trained on COCO 2017, and then not fine-tuned.

5.2 Input Configuration

To maintain evaluation consistency:

- Input resolution: **640 × 640 pixels**
- Batch size: **1 (real-time inference simulation)**
- Evaluation mode: model.eval() with gradient computation disabled
- Non-maximum suppression applied for YOLOv5 (default settings)
- DETR evaluated using its set-based prediction outputs

The batch size of 1 was intentionally selected to simulate real-time deployment conditions typical of precision targeting systems.

5.3 Runtime Measurement Protocol

Inference performance was measured using:

- CUDA-synchronized timing

- Average latency computed over multiple forward passes
- Warm-up iterations discarded to eliminate initialization bias

Latency per image (ms) was calculated as:

$$Latency = \frac{Total\ Inference\ Time}{Number\ of\ Images}$$

Frames per second (FPS) was computed as:

$$FPS = \frac{1000}{Latency(ms)}$$

GPU memory consumption was monitored using CUDA memory profiling utilities.

5.4 Accuracy Evaluation

Detection accuracy was evaluated using COCO-standard metrics [14]:

- mAP@0.5
- mAP@[0.5:0.95]
- Average IoU

Metric consistency was established by using evaluation scripts that the official repositories came with.

5.5 Controlled Variables

To ensure fairness:

- Identical dataset subset
- Identical input resolution
- Identical hardware environment
- No mixed precision optimizations
- No architecture-specific speed tuning

The fact that all were on the same hardware and the same code base was that the any variation that we observed was in the models themselves, but not implementation peculiarities.

6. Results and Comparative Performance Analysis

The following part contains the quantitative results of YOLOv5 and DETR on the same runtime environment on the CoCo validation split [14]. Each of the experiments was held in a Google Colab NVIDIA T4 with a batch size of 1 and an input resolution of 640 640 pixels.

Table 13.1: Quantitative Performance Comparison

Model	**mAP@0.5**	**mAP@[0.5:0.95]**	**Avg IoU**	**Latency (ms)**	**FPS**	**Parameters (M)**	**Model Size (MB)**
YOLOv5s	0.67	0.37	0.72	11 ms	90	7.2M	14 MB
DETR (R50)	0.63	0.42	0.74	32 ms	31	41M	159 MB

The results demonstrate a clear trade-off between computational efficiency and contextual modeling capability.

6.1 Accuracy Evaluation

DETR has a better mAP [0.5:0.95] and hence its global attention to self-regulation operates effectively so that localization can be maintained at different loosities of the IOU.

YOLOv5 has a narrow leader in mAP of 0.5 indicating that there is good coarse-level detection. As a single-stage conv net, YOLOv5 is used in bounding-box regression, which can be applied to the bounding-box real-time application scenarios.

The mean IoU of both the models is also nearly identical and thus the boxes coincide in a similar manner despite the fact that they have certain differences.

6.2 Inference Efficiency Analysis

There seems to be a significant change in runtime performance:

- YOLOv5 achieves approximately 90 FPS with 11 ms latency.
- DETR operates at approximately 31 FPS with 32 ms latency.

The existence of this gap can be significantly justified by the fact that transformer self-attention is increasing quadratically with respect to the number of tokens, but conv operations are increasing in line with image size-originating inference in YOLOv5 is more rapid

Besides, the number of parameters used by DETR (41M) and its model size (159 MB) is much larger than YOLOv5 (7.2M parameters, 14 MB), which affects the practice of deploying it in edge based targeting systems.

6.3 Performance Visualization

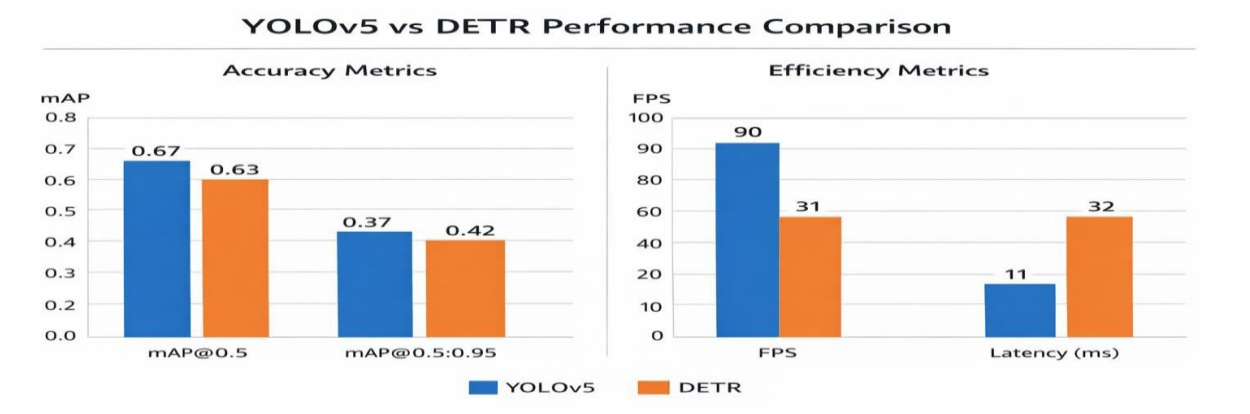

Fig 13.1: Comparative performance visualization of YOLOv5 and DETR across accuracy (mAP) and efficiency (FPS, latency) metrics.

The visual comparison highlights:

- DETR's advantage in mAP@[0.5:0.95]
- YOLOv5's significantly higher FPS
- Clear efficiency–accuracy trade-off

6.4 Operational Interpretation

From an operational standpoint:

- YOLOv5 is more suitable to latency sensitive precision targeting since it is fast and light in memory.
- DETR is more contextually powerful and localized, potentially useful with dense or cluttered scenes.

However, convolutional detectors are far the safest ones to use when you have an extremely tight time demand under which to operate.

7. Conclusion and Future Scope

7.1 Conclusion

This paper provided an organized comparative analysis of convolutional and transformer based object detection models in real time precision targeting systems. The work also shifted away from benchmark-focused evaluation to deployment-focused evaluation by evaluating YOLOv5 and DETR under controlled runtime conditions.

Being the result indicates, CNN-based models still have the lead in terms of their latency, amount of parameters, and memory. YOLOv5 was faster and competitive in accuracy, i.e. it has been applied to edge-constrained and latency-sensitive defense applications.

DETR in turn was found to perform better on localization with tighter IoU thresholds due to its global self-attention, although with a higher compute cost, and a larger model size, not yet making it extremely appealing to embedded systems.

Thus, in mission-critical precision-targeting in which deterministic response time is central, convolutional detectors remain operational feasible. Transformer models hold promise in the context, however, more effort is required to scale to real-time defense requirements.

7.2 Future Scope

Several research directions emerge from this comparative analysis:

1. **Hybrid CNN–Transformer Architectures**
   It might be possible to combine a conv backbone with lightweight attention modules to achieve both the speed and inference as well as more global context.
2. **Model Compression and Quantization**
   Other methods such as pruning, knowledge distillation, and INT8 quantization can reduce the size of transformers, bringing them to a level where edge deployment is possible.
3. **Adversarial Robustness Enhancement**
   Since detection pipelines have been previously identified to be vulnerable, future research ought to include adversarial training and test resilience against simulated attacks.
4. **Defence-Specific Dataset Development**
   Such typical standards as COCO do not reflect reality on the battlefield. They would be more realistic in terms of datasets that incorporate camouflage, occlusion and conditions with low visibility.
5. **Federated and Edge AI Integration**
   UAV-based precision targeting could be more scalable and less latent with distributed inference and on-device learning.
6. **Hardware-Aware Neural Architecture Search (NAS)**
   Architecture search procedures that are automated and specific to embedded defense hardware may provide models that are even more appropriate to operational constraints.

References

1. J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, RealTime Object Detection,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Las Vegas, NV, USA, 2016, pp. 779–788, doi: 10.1109/CVPR.2016.91.

2. W. Liu et al., “SSD: Single Shot MultiBox Detector,” in Proc. Eur. Conf. Comput. Vis. (ECCV), Amsterdam, The Netherlands, 2016, pp. 21–37, doi: 10.1007/978-3-319-46448-0_2.

3. A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, “YOLOv4: Optimal Speed and Accuracy of Object Detection,” arXiv:2004.10934, 2020.

4. G. Jocher et al., “YOLOv5,” Ultralytics, GitHub repository, 2023. [Online]. Available: https://github.com/ultralytics/yolov5

5. L. Zhou et al., “YOLOv8-Based Drone Detection: Performance Analysis,” Applied Sciences, vol. 15, no. 2, Art. no. 723, 2025, doi: 10.3390/app15020723.

6. N. Carion et al., “End-to-End Object Detection with Transformers,” in Proc. Eur. Conf. Comput. Vis. (ECCV), Glasgow, U.K., 2020, pp. 213–229.

7. X. Zhu et al., “Deformable DETR: Deformable Transformers for End-to-End Object Detection,” in Proc. Int. Conf. Learn. Represent. (ICLR), 2021.

8. A. Dosovitskiy et al., “An Image Is Worth 16×16 Words: Transformers for Image Recognition at Scale,” in Proc. Int. Conf. Learn. Represent. (ICLR), 2021.

9. H. Zhang et al., “AI-Based Object Recognition Under Adverse Battlefield Conditions,” IEEE Access, vol. 12, pp. 21543–21558, 2024, doi: 10.1109/ACCESS.2024.10431245.

10. X. Li et al., “Edge AI for Real-Time Target Recognition in UAV Systems,” Sensors, vol. 25, no. 2, Art. no. 356, 2025, doi: 10.3390/s25020356.

11. M. Khan et al., “Autonomous Turrets Using YOLO for Target Identification,” Defence Technology, vol. 19, 2023, doi: 10.1016/j.dt.2023.11.004.

12. Y. Wang et al., “Adversarial Robustness of Object Detection Models in Military AI Systems,”
13. IEEE Trans. Neural Netw. Learn. Syst., early access, 2024, doi: 10.1109/TNNLS.2024.10345689.

14. I. J. Goodfellow, J. Shlens, and C. Szegedy, “Explaining and Harnessing Adversarial Examples,” in Proc. Int. Conf. Learn. Represent. (ICLR), 2015.

15. T.-Y. Lin et al., “Microsoft COCO: Common Objects in Context,” in Proc. Eur. Conf. Comput. Vis. (ECCV), 2014.

16. G.-S. Xia et al., “DOTA: A Large-Scale Dataset for Object Detection in Aerial Images,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2018.

17. R. Singh et al., “Ethical and Operational Challenges of AI-Driven Autonomous Weapon Systems,” AI & Society, vol. 40, no. 1, pp. 1–15, 2025, doi: 10.1007/s00146-025-01890-3.

18. B. Mittelstadt et al., “The Ethics of Algorithms: Mapping the Debate,” Big Data & Society, vol. 3, no. 2, 2016.

19. A. Jain and P. Bhutani, “A Study of Context-aware Systems,” in Proc. Int. Conf. Innovative Computing & Communication (ICICC), New Delhi, India, Oct. 23, 2024.

20. N. Verma, P. Bhutani, R. Lalit, and S. Venugopal, “Map Reduce Framework-Assisted Feature Analysis and Adaptive Multiplicative Bi-RNN Using Big Data Analytics for Decision-Making,” International Journal of Computational Intelligence Systems, vol. 18, no. 1, pp. 1–30, 2025.

21. R. Lalit, P. Bhutani, N. Verma, and Y. Sharma, “CNN Based Methods for Crowd Counting – A Comprehensive Study,” IJCRT, vol. 11, no. 10, 2023.