**International Journal of Emerging Trends in Computer Science and Information Technology**
ISSN: 3050-9246 | https://doi.org/10.63282/3050-9246.IJETCSIT-V4I4P117
Eureka Vision Publication | Volume 4, Issue 4, 159-167, 2023

*Original Article*

# Real-Time Instance Segmentation Using Lightweight CNN-Transformer Hybrids

Sajud Hamza Elinjulliparambil
Pace University.

*Abstract - The problem of instance segmentation is a basic computer vision problem in which localization, classification, and pixel-level localization of individual instances of objects should occur simultaneously. The trade-off between computational performance and segmentation quality is a big challenge in achieving real-time performance in instance segmentation especially in resource-constrained platforms like edge devices and embedded systems. Hybrid architecture Lightweight CNN Transformer hybrid networks have become an exciting solution, which combines the ability to extract local features efficiently of convolutional networks with the ability to model the global context of Transformers. This literature review entails an in-depth examination of the instant segmentation methods in real-time, with the focus on CNN-based pipelines, Transformer models, and their hybrid versions. Lightweight design strategies, such as model compression, efficient attention mechanisms, and backbone optimization, and standard datasets and benchmarking protocols to ensure consistent evaluation are discussed. Lastly, we look at real-life implementations in autonomous driving, robotics, and in industry vision, and determine current challenges and future research opportunities in accuracy, efficiency and robustness in real-time implementation.*

*Keywords - Instance segmentation; real-time vision; CNN Transformer hybrid; attention mechanism; model efficiency; autonomous system; robotics; computer vision.*

## 1. Introduction

The instance segmentation is a fundamental issue in computer vision that seeks to localize, classify and simultaneously delineate individual objects at the pixel level [1]. In contrast to semantic segmentation, which labels the pixels with the help of the class, but does not determine object occurrences, and object detection, which only coarsely localizes the objects with the help of the bounding box, instance segmentation offers object-specific understanding of the objects by creating a separate segmentation mask per object. This ability is important in applications that need accurate interpretation of a scene, e.g., autonomous systems, robotic perception, and intelligent surveillance, in which object identity and object shape information are required.

Real time instance segmentation builds on this task by adding very demanding computational constraints, such as high frame rates, low latency and allowed memory on the model of streaming visual information [2]. In edge and embedded platforms in particular, these limitations are amplified by the fact that the amount of available computational resources and power constraints are by their very nature restricted. This has led to the fact that the objectives of finding high accuracy of segmentation and real-time performance have become the focus of research.

The initial segmentation of the instances was based on the development of the convolutional neural network that was initially developed to classify images and detect objects [3]. Two-stage pipelines detect candidate object regions, and then run the prediction of the pixel-level mask of every case, being highly accurate but with significant computational penalties. To overcome efficiency issues, one-stage and prototype-based systems were proposed, which would allow inference to be made much faster by decoupling object generation and instance classification or object-to-object sharing of computation.

Even with those advances, purely convolutional models still do not perform well in modelling long-range spatial dependencies, which are the key in the correct segmentation of objects in the presence of occlusions, or in complex environments [4]. Meanwhile, real-time deployment imposes further limits on latency, frame per second throughput rate and memory footprint. These conflicting demands drive the search to explore novel structures that are able to embody both a regional spatial information as well as global contextual relationship, without having to pay prohibitive computational costs. The instance segmentation in real time is subject to a number of issues that are interconnected [5]. One of those is the accuracy versus efficiency trade-off as the model capacity is often increased with the result of better segmentation quality and worse inference speed. Models with lightweight design can have issues with finely delineating boundaries, small instances of objects or highly obscured scenes.

Another problem is multi-scale representation of objects because in the real world, there is often a significant variation in the size of objects. Though feature pyramid structures reduce scale variation, they add to the complexity of computation. Occlusion and thick object layouts can also make the separation of the instances more difficult, and thus it demands effective contextual reasoning to distinguish between the overlapping instances. Besides this, the complexity of the models that can be implemented in real world practice is limited by hardware constraints including limited memory bandwidth and parallel processing power.

As a solution to the weaknesses of convolutional architectures to capture global context, attention-based mechanisms and Transformer models were brought to the visual recognition problems [6]. Transformers provide a highly effective tool of modeling long-range dependencies by self-attention, making it possible to perform better relational reasoning on spatial regions. Nevertheless, direct application of Transformers to dense prediction issues entails a high computational cost, and practicality of Transformer-only solutions to real time instance segmentation with limited resources is not possible [7].

This drawback has given rise to hybrid architectures that unite convolutional neural networks with the Transformer parts. Convolutional backbone is effective in such designs, which extract local and multi-scale features, and lightweight attention or Transformer modules increase the global context modeling. Combining these two strengths complements, CNN-Transformer hybrids attain a more optimal accuracy/efficiency balance, and therefore are effective in real-time instance segmentation on hardware-limited systems.

Figure 1 demonstrates the conceptual history of architecture of instance segmentation, showing how the all-convolutional pipelines have evolved into attention-based and hybrid models. The development highlights how efficiency-based CNN models were progressively extended with the global context modeling elements to counteract the increasing complexity of the real-time segmentation problems and preserve realistic inference perceptions[8].

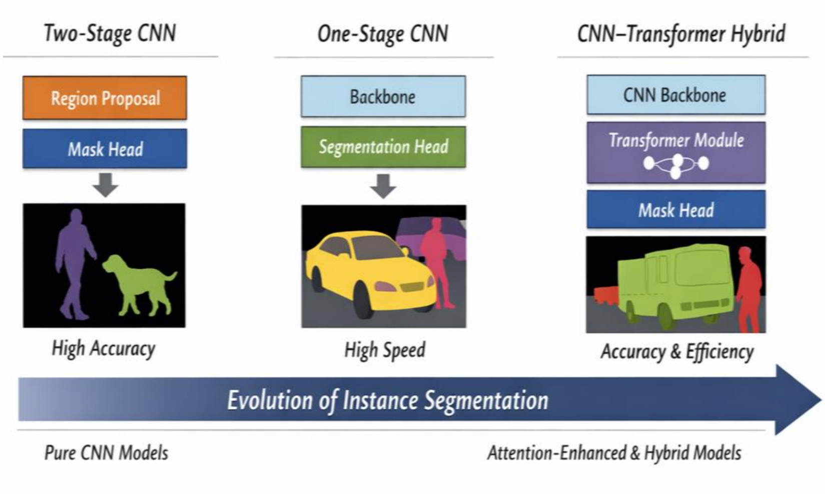

**Figure 1. Evolution of Instance Segmentation Architectures**

The objective of this review is to analyze real-time instance segmentation methods with an emphasis on lightweight CNN–Transformer hybrid architectures suitable for resource-constrained environments. The review aims to summarize key architectural design principles, highlight efficiency-oriented strategies, and synthesize existing approaches that balance segmentation accuracy with real-time performance. Additionally, it seeks to identify common challenges and research gaps related to deployment, scalability, and robustness.

## 2. Fundamentals of Instance Segmentation

The instance segmentation is a dense prediction challenge, whose main goal is to integrate object localization, classification, and segmentation on a pixel level into one framework [9]. It is a basic constituent of fine grained visual comprehension and the conceptual underpinning of the architectural evolution later to be mentioned.

### *2.1. Definition and Problem Formulation*

The instance segmentation goal is to identify and divide each instance of an object in an image on its own [10]. The model generates a set of outputs based on class labels, localization of these classes and pixel-wise masks of a given input image. The masks are associated with different object instances, despite the fact that a number of objects may fall in the same semantic group.

Officially, the task may be considered a prediction of a variable-length list of instances, with each instance being represented by a 4-tuple of the form (class label, bounding box or localization descriptor of the same type, binary or probabilistic segmentation mask). This definition is to contrast instance segmentation with semantic segmentation, which does not divide discrete object instances, and object detection, which is deprived of detailed shape information [11].

### *2.2. Evaluation Metrics*

The instance segmentation models are usually evaluated on the basis of the metrics that simultaneously quantify the accuracy of the localization, classification accuracy, and quality of masks [12] . One of the most commonly used accuracy measures is Mean Average Precision which is calculated by comparing predicted masks to ground truth annotations with a series of overlap thresholds. Such variants as AP50 and AP75 can give us more understandable performance indicators at certain overlap criteria [13].

In the case of applications that need real-time, the accuracy metrics are not sufficient. Computational efficiency is also of the essence and is usually considered in terms of inference speed (usually in terms of number of frames per second), end to end latency and memory footprint. These measures indicate the appropriateness of a model to be used in time-sensitive and resource-limited situations.Table 1 reveals a summary of typical evaluation measures of the instance segmentation in real-time, the difference between accuracy-based and efficiency-based measures.

**Table 1. Common Evaluation Metrics for Real-Time Instance Segmentation**

| Category | Metric | Brief Description |
|---|---|---|
| Accuracy | mAP | Overall segmentation accuracy across overlap thresholds |
| Accuracy | AP50 / AP75 | Precision at coarse and strict mask overlap levels |
| Efficiency | FPS | Inference speed indicating real-time capability |
| Efficiency | Latency | Time per image during inference |
| Efficiency | Memory | Runtime memory usage for deployment |

### *2.3. Real-Time Constraints and Deployment Scenarios*

The real-time instance segmentation systems are forced to work within very stringent requirements, dictated by real-life deployment conditions [14]. In intelligent transportation systems and autonomous driving, visual streams at high-resolution are demanded by the models with minimum latency to enable the models to make timely decisions. Robotic perception systems require sound instance level perception to facilitate object control and navigation over dynamic surroundings.

These constraints are further exacerbated by mobile and embedded vision applications, which are low in terms of computational power and data requirements, and memory memory. In those, models should have a delicate balance between the ability to represent and efficiency. Such deployment assumptions have a severe impact on the architectural design decisions and make it difficult to develop lightweight or hybrid segmentation models that can provide real-time performance with an insignificant accuracy degradation.

## 3. CNN-Based Real-Time Instance Segmentation

CNN-based approaches were foundational in real-time instance segmentation, providing efficient feature extraction and mask prediction capabilities. These methods form the baseline for hybrid CNN–Transformer architectures and prioritize inference speed without severely compromising accuracy [15].

### *3.1. Two-Stage CNN Approaches*

In two-stage algorithms, proposals are created in the first stage and class labels and instance masks of each region are predicted in the second stage.

#### *3.1.1. Mask R-CNN and Lightweight Variants*

Mask R-CNN is an extension of Faster R-CNN provided with an additional parallel branch to predict pixels masks. Lightweight models minimize the complexity of the backbones or use lower-complexity region proposal networks, which are faster to inference but have moderate accuracy [16].

#### *3.1.2. Restrictions to Real Time Usage.*

Two-stage pipelines are also computationally intensive and therefore less suitable in edge devices and embedded systems despite their high accuracy leading to lower FPS and higher latency [17].

### *3.2. One-stage and Prototyping-based Approaches.*

One stage and prototype based techniques do not have the region proposal step to enhance efficiency.

#### *3.2.1. YOLACT and YOLACT++*

These models produce a collection of prototype masks and the associated per-instance coefficients during one forward pass, making such models yield inference more quickly with competitive segmentation quality [18].

*3.2.2. SOLO and Related Approaches*

SOLO develops an instance segmentation model as a pixel-wise classification problem on a spatial grid which makes calculations simpler, as well as faster than two-stage models.

### *3.3. Lightweight CNN Backbones*

The main consideration of the backbone is to choose a suitable backbone that will perform well in real time.

- MobileNet: Uses depthwise separable convolutions to minimize parameters and calculations whilst maintaining feature representation.
- ShuffleNet: Uses group convictions and channel shuffling to enhance the efficiency and the accuracy.
- EfficientNet: Scales the depth, width, and resolution of a network with compound scaling, having good performance at low computational cost.

A typical CNN-based real-time instance segmentation pipeline is shown in Figure 2. It emphasizes the backbone feature extraction, multi-scale feature pyramid, and the mask prediction head which jointly allow making predictions on the instance level efficiently.

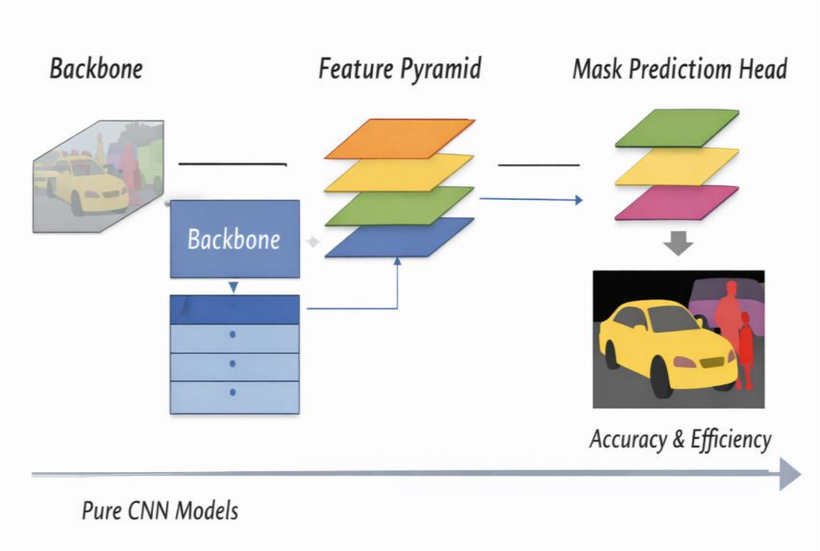

**Figure 2. Representative Cnn-Based Real-Time Instance Segmentation Pipeline**

## 4. Vision Transformers in Segmentation

Transformers which were first created in natural language processing have also been expanded to computer vision to address the locality restrictions of convolutional neural networks [19]. Their self attention response allows long-range dependencies to be modeled, which is particularly applicable in the instance segmentation of complex or covered scenes. Vision Transformers (ViT) and other architectures proved that attention-based models could compete with CNNs at image understanding tasks, however, the aspect of computation practice is still an issue [20].

### *4.1. Self-Attention and Global Context Modeling*

The computation of the relationships between all the elements of a feature map by self-attention results in a context-aware representation that incorporates the information of the whole image [21]. This enables the model to record the interactions of far away pixels which increases the segmentation of intersecting objects and provides consistency of large-scale structures. In contrast to convolutional layers, which are local information aggregating layers, self-attention offers a global reasoning mechanism at every network layer.

### *4.2. Early Vision Transformers*

*4.2.1. Vision Transformer (ViT)*

ViT works using images which have been split into fixed-size patches, flattened into tokens, and then fed into blocks of Transformer. The model can exploit the self-attention mechanism to have long-range interaction of features and show competitive performance on the classification and segmentation tasks when trained on large datasets.

*4.2.2. Detection Transformer (DETR)*

DETR approaches object detection and instance segmentation as a set prediction problem, where an encoder-decoder Transformer architecture is used [22]. It does not require post-processing tasks, such as non-maximum suppression and anchor design, which have to be done by hand, and it directly causes instance-level predictions using global image features.

### *4.3. Limitations of Pure Transformer Models for Real-Time Use*

Although Transformers are effective in global context learners, it comes with a large computation cost. Self-attention has quadratic complexity in relation to the number of tokens, and hence high-resolution processing of inputs is costly. Transformers are also generally not able to attain performance unless pretrained on large datasets and their memory footprint can be prohibitive to embedded or edge devices. These limitations make them less practical to apply in real-time, which prompts hybrid architecture with CNN performance and Transformer-based global reasoning.

Table 2 gives a brief comparison of CNNs and Transformers in instance segmentation and outlines their advantages and disadvantages.

**Table 2. CNN vs. Transformer Comparison for Segmentation**

| Aspect | CNN | Transformer |
|---|---|---|
| Receptive Field | Local, grows with depth | Global from first layer via self-attention |
| Feature Representation | Hierarchical, spatially local | Global context-aware, captures long-range dependencies |
| Efficiency | High, lower memory and computation | Moderate to low, high memory and compute cost |
| Scalability | Easier for high-resolution inputs | Challenging due to quadratic attention complexity |
| Real-Time Suitability | High, can be deployed on edge devices | Limited, computationally heavy for real-time use |

Table 2 summarizes qualitative differences between CNNs and Transformers in segmentation tasks. CNNs are efficient and suitable for real-time applications but limited in global reasoning, whereas Transformers provide rich global context but are computationally intensive and less practical for high-speed deployment.

## 5. CNN–Transformer Hybrid Architectures

CNN–Transformer hybrid architectures emerged to combine the efficiency of convolutional networks with the global context modeling capabilities of Transformers [23]. These architectures aim to maintain real-time performance while improving segmentation accuracy, particularly in challenging scenarios involving occlusion, small objects, or cluttered scenes. Complementary lightweight strategies further reduce computation and memory footprint, enabling deployment on edge and resource-constrained devices.

### *5.1. Design Philosophy of Hybrid Models*

The principle of the design of hybrid models is to use CNNs to extract local, multi-scale features and Transformers to extract global spatial dependencies. CNN backbones are effective at processing high-resolution data and representation of features hierarchically, but Transformer modules offer long-range relational reasoning and are better at separating instances and maintaining consistency of masks.

### *5.2. Encoder-Decoder Hybrid Architectures.*

The architecture of hybrid encoder decoder is based on a CNN encoder to generate feature maps at various resolutions, which are inputted into Transformer-based decoders or attention blocks [24]. In the refining of instance masks, the decoder utilizes global context, which improves performance on overlapping or complex objects without significantly changing the computational expense.

### *5.3. Attention-Augmented CNNs*

Other hybrid designs, typically not considered full Transformers, incorporate lightweight attention modules, including non-local blocks or axial attention, directly into CNN layers. The modules to these features enhance convolutional features with global information that boosts the quality of segmentation and also maintains inference efficiency.

### *5.4. Strategy tokenization and fusion of features.*

Multi-scale feature pyramids, patch embeddings and tokenization strategies are used to enable hybrid architectures to effectively combine local features with global features. CNN-based features are divided into tokens or merged at different scales so that Transformers can make decisions on a global scale but retain spatial information at high-resolution.Figure 3 illustrates a generic CNN–Transformer hybrid architecture for instance segmentation. The diagram shows a CNN backbone extracting multi-scale features, feeding them into a lightweight Transformer module, and producing instance masks through a dedicated prediction head.

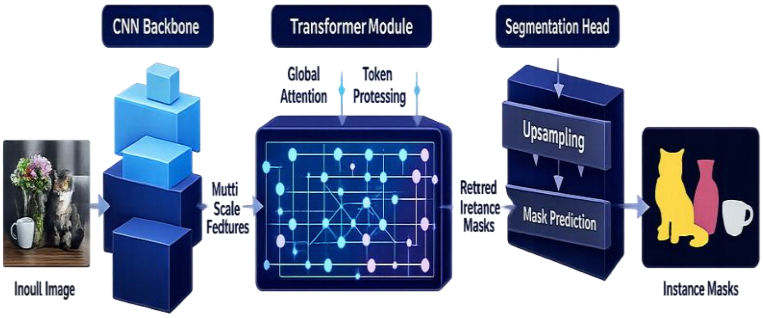

**Figure 3. Generic CNN–Transformer Hybrid Pipeline**

## 6. Lightweight Design Strategies and Datasets for Real-Time Instance Segmentation

The deployment of CNN -Transformer hybrids in real-time depends not only on the architecture design but also on the approaches to minimise the computation and memory footprint. It is also important to test these models on standard datasets and benchmarking protocols in order to have uniform performance testing.

### *6.1. Model Compression and Reduction of Parameters.*

Pruning, weight sharing, and small attention can be used in hybrid models to enhance the efficiency of inference. These methods minimize the overall parameter and operations count and retain the majority of the accuracy of the segmentation and make models more appropriate to edge devices.

### *6.2. Efficient Attention Mechanisms.*

Attention mechanisms include linear attention, sparse attention and window-based attention, which are approximate global context models at much lower computational cost compared to standard self-attention. These are ways of keeping the advantages of global reasoning and at the same time inference is much faster and consumes less memory.

### *6.3. Backbone Optimization of Edge Devices.*

By combining lightweight CNN backbones such as MobileNet, ShuffleNet, or EfficientNet with Transformer modules, individual image processing can be performed in real-time and does not surpass memory or compute limits. Additional strategies of accuracy and efficiency include multi-scale feature fusion and cautious tokenization. Table 3 provides a summary of lightweight strategies which have been widely used in CNN-Transformer hybrids, with which component each is impacted and what efficiency advantage.

**Table 3. Lightweight Strategies in CNN–Transformer Hybrids**

| Strategy | Affected Component | Efficiency Benefit |
|---|---|---|
| Pruning | CNN/Transformer weights | Reduces parameter count and computation |
| Weight Sharing | Attention modules | Lowers memory usage without reducing context modeling |
| Compact Attention | Transformer blocks | Maintains global reasoning at lower cost |
| Linear/Sparse/Window Attention | Attention layers | Reduces quadratic complexity of self-attention |
| Lightweight Backbones | CNN feature extractor | Faster inference and lower memory footprint |
| Multi-Scale Feature Fusion | Tokenization / Pyramids | Balances accuracy with efficiency |

Table 3 summarizes key strategies used to make CNN–Transformer hybrids suitable for real-time deployment, highlighting affected components and efficiency improvements.

### *6.4. Standard Datasets*

The evaluation of instance segmentation methods is consistently important, which depends on benchmarking and dataset selection. COCO and Cityscapes are the most popular datasets of large-scale natural images and urban scene understanding, respectively [25][26]. These datasets are of high quality with annotations of instance masks, bounding boxes, and class labels, which allow the severe evaluation of performance.

### *6.5. Real Time Evaluation Protocols.*

Accuracy measures are not the only metrics used to analyze real-time performance, but computational efficiency is also used. The standard protocols are used to measure frames per second ( FPS ), delay per image and memory consumption under certain hardware conditions. Regular benchmarking guarantees reasonable comparison of techniques, especially with lightweight models that are to be used in edge deployment.

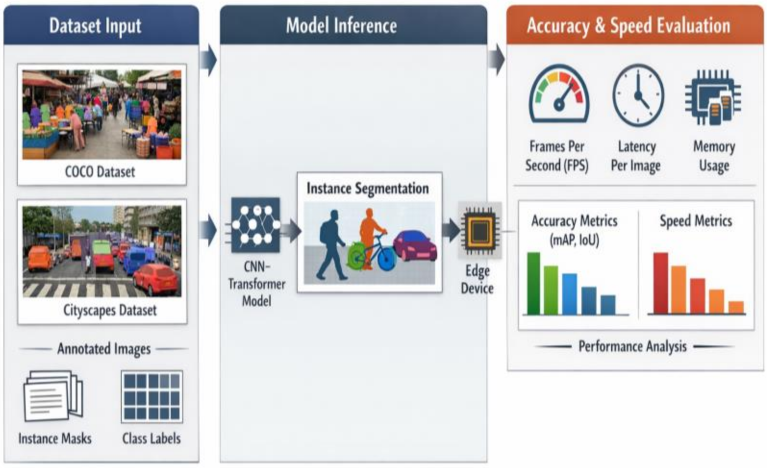

**Figure 4. Benchmarking Workflow for Real-Time Instance**

Figure 4 illustrates a standard benchmarking workflow for real-time instance segmentation, including dataset input, model inference, and evaluation of both accuracy and speed metrics.

## 7. Applications of Real-Time CNN–Transformer Hybrid Segmentation

CNN Transformer hybrid models have seen applications in a wide range of real-time vision tasks, especially in problems where instance-level perception (being fast and accurate) is of critical importance. They are appropriate to resource-constrained settings requiring fast inference as the combination of local feature extraction and global context modeling is well-suited.

### *7.1. Self-Driving cars and Smart transportation.*

Instance segmentation is a vital part of autonomous driving as it can analyze and comprehend complex traffic situations, both vehicles, pedestrians, and cyclists, and other objects in real-time. CNNTransformer hybrids Scene understanding CNNTransformer hybrids are able to effectively utilize efficient feature extraction coupled with the reasoning of global context to enable accurate detection and segmentation despite scenes under occlusion or even under the conditions in a traffic jam. Real-time performance provides timeliness in the decision-making throughout the navigation and safety-critical systems.

### *7.2. Robotics and Embedded Vision.*

Low-latency and high-precision instance segmentation is required in robotic perception systems used in object manipulation, grasping, and navigation in dynamic environments. Hybrid models enable robots to infer high-resolution images in a short time and fine-grained segmentation accuracy, which makes them applicable to be deployed to embedded system architectures with limited computational capabilities.

### *7.3. Medical and industrial vision (real time scenarios).*

Fast and accurate isolation of singular objects or structures is usually required in medical imaging and industrial inspection. Hybrid CNN-Transformer models are also studied to be used in detection of surgical tools or segmentation of organs/tissues or even defect detection in a manufacturing line, where processing needs to be in real-time and has to guide an intervention or an automated process without loss of accuracy.

## 8. Challenges and Future Directions

Although there are significant improvements in real-time CNNs-Transformer hybrid instance segmentation, there are still several issues that are yet to be addressed. The accuracy - efficiency tradeoff is a big problem: to get high quality segmentation, it might be necessary to incur more computation and memory usage, which is problematic on resource-constrained devices. One of the important design considerations remains speed and precision in segmentation.

Scalability to high-resolution inputs is also another major limitation. When global context modelling is modeled in terms of attention mechanisms, processing large images or video streams may result in a memory bottleneck and significant overhead in computational load. Multi-scale features and high-resolution representation are fundamental research issues that are handled efficiently.

There are also problems of generalization and robustness. Trained models on the particular data set cannot be useful in case of domain shifts or when they are applied to unseen scenes and the availability of instance-level annotated data is limited, which prevents the wider applicability. The ability to resist occlusions, changes in lighting and interactions between multiple complex objects is of considerable significance to real-time applications in dynamic environments.

In the future, there are various paths that can be followed according to the trends. It is possible to design more efficient hybrid attention mechanisms with less computational cost but still have abilities to model global context. Initial studies on unified detection segmentation models suggest that it is possible to simplify pipelines by integrating object detection and instance segmentation in a unified, end to end system. Lastly, hardware-conscious design and Edge-AI-oriented approaches such as neural architecture search and co-design of models taking device constraints into account are likely to be important contributors towards supporting real-time instance segmentation on embedded and edge devices.

These issues and views show that there is still a necessity in increasing model efficiency, robustness, and deployment-conscious design, which gives direction to the researchers who aim at the further development of real-time instance segmentation.

## 9. Conclusion

Real-time instance segmentation is a major but difficult challenge in computer vision, with the models needed to provide accurate object delineation with a severe computational budget. Conventional CNN-based methods formed the basis of effective feature extraction and mask prediction, however, their lower ability to capture long-range interactions prohibits performance in complicated and occluded scenes. With the introduction of Vision Transformers, the use of global context models is now powerful due to self-attention mechanisms and enhances the quality of relational reasoning and segmentation, but the computational cost of bare Transformer models is high and restricts their use in real-time applications.

Hybrid CNN Transformer architectures can be effectively used to fill this gap as they are efficient at using convolutional backbones with the global reasoning of lightweight attention or Transformer modules. These hybrids also provide a desirable trade-off between speed and accuracy and are, therefore, well-suited to edge and embedded devices where memory, power, and latency are the primary factors. The major design solutions, such as optimization of the backbone, feature fusion on a larger scale, compact attention, and model compression, further improve the real-time performance without significantly affecting the segmentation quality. Its results on standard datasets, e.g. COCO and Cityscapes, and strict evaluation guidelines demonstrate the relevance of the approaches in a wide variety of real-world conditions.

CNN transformer hybrids have been applied in autonomous driving, robotic perception, and real-time industrial and medical vision and have proven to be capable of operating in dynamically changing, high-resolution environments and achieve precise instance-level understanding. Nevertheless, several issues with achieving a balance between accuracy and efficiency, scaling to high-resolution inputs and robustness to domain shifts, occlusions and complex interaction of objects have yet to be resolved. Future studies will probably center on more effective attention systems, single- detection segmentation models and hardware-compatible designs to edge-AI requirements.

As a whole, lightweight CNN-Transformer hybrids are an exciting future in real-time instance segmentation to provide a compelling trade off between speed, accuracy, and contextual reasoning capable of meeting the requirements of today’s vision-based application and future technologies in the development of efficient, deployable segmentation systems.

## References

[1] A. M. Hafiz and G. M. Bhat, “A survey on instance segmentation: State of the art,” Int. J. Multimedia Inf. Retrieval, vol. 9, no. 3, pp. 171–189, 2020.

[2] D. Bolya, C. Zhou, F. Xiao, and Y. J. Lee, “YOLACT: Real-time instance segmentation,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Seoul, South Korea, 2019, pp. 9157–9166.

[3] R. Yang and Y. Yu, “Artificial convolutional neural network in object detection and semantic segmentation for medical imaging analysis,” Frontiers in Oncology, vol. 11, Art. no. 638182, 2021.

[4] Z.-M. Chen, X.-S. Wei, P. Wang, and Y. Guo, “SST: Spatial and semantic transformers for multi-label image recognition,” IEEE Trans. Image Process., vol. 31, pp. 2570–2583, 2022.

[5] Eshed Ohn-Bar and M. M. Trivedi, “Hand gesture recognition in real time for automotive interfaces: A multimodal vision-based approach and evaluations,” IEEE Trans. Intell. Transp. Syst., vol. 15, no. 6, pp. 2368–2377, 2014.

[6] Y. Li, K. Yang, W. Chen, and Y. Li, “Contextual transformer networks for visual recognition,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 45, no. 2, pp. 1489–1500, 2022.

[7] J. Yang, C. Fan, H. Wang, Y. Wang, and B. Chen, “Focal attention for long-range interactions in vision transformers,” in Adv. Neural Inf. Process. Syst. (NeurIPS), vol. 34, 2021, pp. 30008–30022.

[8] S. Mehtab, Deep Neural Networks for Road Scene Perception in Autonomous Vehicles Using LiDARs and Vision Sensors, Ph.D. dissertation, Auckland Univ. of Technology, Auckland, New Zealand, 2022.

[9] P. Luc, C. Couprie, S. Chintala, and J. Verbeek, “Predicting future instance segmentation by forecasting convolutional features,” in Proc. Eur. Conf. Comput. Vis. (ECCV), Munich, Germany, 2018, pp. 584–599.

[10] J. Cao et al., “D2Det: Towards high quality object detection and instance segmentation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Seattle, WA, USA, 2020, pp. 11485–11494.

[11] L.-C. Chen et al., “MaskLab: Instance segmentation by refining object detection with semantic and direction features,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Salt Lake City, UT, USA, 2018, pp. 4013–4022.

[12] S. M. Harrison, L. G. Biesecker, and H. L. Rehm, “Overview of specifications to the ACMG/AMP variant interpretation guidelines,” Curr. Protoc. Hum. Genet., vol. 103, no. 1, Art. no. e93, 2019.

[13] David DeBonis et al., “APE: Metrics for understanding application performance efficiency under power caps,” Sustainable Computing: Informatics and Systems, vol. 34, Art. no. 100702, 2022.

[14] A. M. Shabut et al., “An intelligent mobile-enabled expert system for tuberculosis disease diagnosis in real time,” Expert Syst. Appl., vol. 114, pp. 65–77, 2018.

[15] W. Wang, H. Lin, and J. Wang, “CNN-based lane detection with instance segmentation in edge-cloud computing,” J. Cloud Comput., vol. 9, no. 1, Art. no. 27, 2020.

[16] J. Park and H. Moon, “Lightweight Mask R-CNN for warship detection and segmentation,” IEEE Access, vol. 10, pp. 24936–24944, 2022.

[17] B. Kim et al., “Energy-efficient acceleration of deep neural networks on real-time-constrained embedded edge devices,” IEEE Access, vol. 8, pp. 216259–216270, 2020.

[18] C. Zhou, YOLACT++: Better Real-Time Instance Segmentation, Univ. of California, Davis, CA, USA, Tech. Rep., 2020.

[19] M. Ekman, Learning Deep Learning: Theory and Practice of Neural Networks, Computer Vision, Natural Language Processing, and Transformers Using TensorFlow, Boston, MA, USA: Addison-Wesley, 2021.

[20] S. Khan et al., “Transformers in vision: A survey,” ACM Comput. Surveys, vol. 54, no. 10s, pp. 1–41, 2022.

[21] B. Yang et al., “Context-aware self-attention networks,” in Proc. AAAI Conf. Artif. Intell., vol. 33, no. 1, 2019, pp. 9334–9341.

[22] N. Carion et al., “End-to-end object detection with transformers,” in Proc. Eur. Conf. Comput. Vis. (ECCV), Cham, Switzerland: Springer, 2020, pp. 213–229.

[23] R. Shao, X.-J. Bi, and Z. Chen, “A novel hybrid transformer-CNN architecture for environmental microorganism classification,” PLOS ONE, vol. 17, no. 11, Art. no. e0277557, 2022.

[24] C. Zhang et al., “Transformer and CNN hybrid deep neural network for semantic segmentation of very-high-resolution remote sensing imagery,” IEEE Trans. Geosci. Remote Sens., vol. 60, pp. 1–20, 2022.

[25] H. Caesar, J. Uijlings, and V. Ferrari, “COCO-Stuff: Thing and stuff classes in context,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Salt Lake City, UT, USA, 2018, pp. 1209–1218.

[26] M. Cordts et al., “The Cityscapes dataset for semantic urban scene understanding,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Las Vegas, NV, USA, 2016, pp. 3213–3223.