IJIRSET
International Journal of Innovative Research in
SCIENCE | ENGINEERING | TECHNOLOGY

e-ISSN: 2319-8753 | p-ISSN: 2347-6710


# INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

## IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 13, Issue 5, May 2024

ISSN INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.423

9940 572 462 | 6381 907 438 | ijirset@gmail.com | www.ijirset.com

International Journal of Innovative Research in Science, Engineering and Technology (IJIRSET)

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Refereed Journal |

|| Volume 13, Issue 5, May 2024 ||

| DOI:10.15680/IJIRSET.2024.1305321 |

# Multiple Object Detection and Re-Identification Using Detectron2

**Raji Krishnan R, Athira T P, Mahesh Murali**

P G Student, Department of MCA, Mount Zion College of Engineering, Pathanamthitta, India

Assistant Professor, Department of MCA, Mount Zion College of Engineering, Pathanamthitta, India

Assistant Professor, Department of MCA, Mount Zion College of Engineering, Pathanamthitta, India

**ABSTRACT**: Multi-object tracking (MOT) plays a crucial role in computer vision applications, including action recognition, surveillance, and autonomous driving. While detection-based trackers have shown impressive performance in MOT, they often struggle in challenging scenarios where objects overlap or separate. In this project, we propose an innovative approach that combines Detectron2, triplet-based MOT networks, and attention-based re-identification models to enhance multi-object tracking

**KEYWORDS**: Detectron2, Mask R-CNN, Multiple object detection, Re-identification, ImageSegmentation

## I. INTRODUCTION

This project aims to tackle the complex task of multiple object detection by leveraging state-ofthe-art deep learning techniques. By harnessing the power of convolutional neural networks (CNNs) and advanced object detection architectures such as YOLO (You Only Look Once) and Faster R-CNN (Region-based Convolutional Neural Networks), we endeavor to develop a robust and efficient system capable of detecting various objects with high accuracy and speed. The outcomes of this project have the potential to significantlyimpact numerous industries, from enhancing safety in autonomous vehicles to optimizing inventory management in retail settings. By pushing the boundaries of multiple object detection, we aim to contribute to the advancement of computer vision technology and pave the way for innovative applications that improveefficiency, safety, and convenience in various domains.

Multiple-object tracking (MOT), as a high-level task in computer vision, has been widely applied in various fields such as video surveillance , human behavior recognition, and autonomous driving [One of the advantages of video-based MOT is its ability to accurately localize targets continuously in videos while maintaining their identity information, unchanged, when the appearance or the surrounding environment changes. This allows for the output of complete motion trajectories for tracked objects. As a result, it has gained increasing attention from researcher.

Detectron2 is a ground-up rewrite of Detectron that started with maskrcnn-benchmark. The platform is now implemented in PyTorch . With a new, more modular design, Detectron2 is flexible and extensible, and ableto provide fast training on single or multiple GPU servers. Detectron2 includes high-quality implementations of state-of-the-art object detection algorithms, including DensePose, panoptic feature pyramid networks, and numerous variants of the pioneering Mask R-CNNmodel family also developed by FAIR. Its extensible design makes it easy to implement cutting-edge research projects without having to fork the entire codebase.

## II. RELATED WORK

Object Detection is a computer vision task in which the goal is to detect and locate objects of interest in an image or video. The task involves identifying the position and boundaries of objects in an image, and classifying the objects into different categories. It forms a crucial part of vision recognition, alongside image classification and retrieval. Over the years, various methods have been developed to tackle this problem.

Feature detection includes methods for computing abstractions of image information and making local decisions at every image point whether there is an image feature of a given type at that point or not. The resulting features will be subsets of the image domain, often in the form of isolated points, continuous curves or connected regions.Once features have been detected, a local image patch around the feature can be extracted. This extraction may involve quite considerable amounts of image processing. The result is known as a feature descriptor or feature vector. Among the

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Refereed Journal |

|| Volume 13, Issue 5, May 2024 ||

| DOI:10.15680/IJIRSET.2024.1305321 |

approaches that are used to feature description, one can mention N-jets and local histograms (see scale-invariant feature transform for one example of a local histogram descriptor). In addition to such attribute information, the feature detection step by itself may also provide complementary attributes, such as the edge orientation and gradient magnitude in edge detection and the polarity and the strength of the blob in blob detection.

- Non-Neural Approaches: Viola–Jones Object Detection: This framework is based on Haar features and hasbeen widely used for face detection.
- Scale-Invariant Feature Transform (SIFT): SIFT extracts distinctive features from images and iscommonly used for object recognition.
- Neural Network Approaches: Region Proposals (R-CNN, Fast R-CNN, Faster R-CNN): These methods use region proposals to identify potential object locations and then classify them usingCNNs.

- Deep Learning-Based Approaches: Fast-RCNN, YOLOv3, and FCOS are examples of detection modelsbuilt on datasets like ImageNet, COCO, and PASCAL VOC

The R-CNN detector first generates region proposals using an algorithm such as Edge Boxes. The proposal regions are cropped out of the image and resized. Then, the CNN classifies the cropped and resized regions. Finally, the region proposal bounding boxes are refined by a support vector machine (SVM) that is trained usingCNN features.

As in the R-CNN detector , the Fast R-CNN detector also uses an algorithm like Edge Boxes to generate region proposals. Unlike the R-CNN detector, which crops and resizes region proposals, the Fast R-CNN detector processes the entire image. Whereas an R-CNN detector must classify each region, Fast R-CNN pools CNN features corresponding to each region proposal. Fast R-CNN is more efficient than R-CNN, because in the Fast R-CNN detector, the computations for overlapping regions are shared.The Faster R-CNN detector adds a region proposal network (RPN) to generate region proposals directly in the network instead of using an external algorithm like Edge Boxes. The RPN uses Anchor Boxes for Object Detection. Generating region proposals in the network is faster and better tuned to your data.

## III. PROPOSED SYSTEM

Detectron2 Integration: Detectron2 is a state-of-the-art object detection library. It provides a robust framework for detecting objects in images. By integrating Detectron2 into the system, you'll have a powerful tool for accurately detecting objects, which serves as the initial step in the multi-object tracking process.

Triplet-based MOT Networks: Triplet-based MOT networks are used for associating detections across frames. These networks learn embeddings for each detected object and use distance metrics, such as cosine distance, to determine associations between objects in consecutive frames. By utilizing triplet-based MOT networks, the system can effectively handle scenarios where objects overlap or separate.

Attention-based Re-identification Models: Re-identification models are crucial for correctly identifying and re-associating objects that may have been lost or occluded during tracking. Attention-based mechanisms can help focus on relevant features for re-identification, improving the accuracy and robustness of the tracking system, especially in complex scenes.

Integration and Fusion: The proposed system integrates these components seamlessly to create a robust multi- object tracking system. Detectron2 provides accurate object detections, which are then fed into the triplet-based MOT networks for association across frames. The attention-based re-identification models further refine the tracking by focusing on relevant features and improving object re-identification.

Challenging Scenario Handling: By combining these techniques, the system aims to overcome challenges commonly encountered in multi-object tracking, such as object occlusion, separation, and varying appearances. The triplet-based networks and attention mechanisms enhance the system's ability to accurately track objects in complex scenarios, making it suitable for applications like action recognition, surveillance, and autonomous driving.

Performance Evaluation: The proposed system should be evaluated extensively on benchmark datasets for multi-object tracking to assess its performance compared to existing methods. Metrics such as MOTA (Multiple Object Tracking Accuracy) and IDF1 (ID F1 Score) can be used to quantify the tracking accuracy and robustness of the system across

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Refereed Journal |

|| Volume 13, Issue 5, May 2024 ||

| DOI:10.15680/IJIRSET.2024.1305321 |

different scenarios.

By combining detection-based tracking with triplet-based MOT networks and attention-based re-identification models, the proposed system aims to achieve state-of-the-art performance in multi-object tracking, especially in challenging scenarios encountered in real-world applications.

## III. METHODOLOGY

Detectron2 is a ground-up rewrite of Detectron that started with maskrcnn-benchmark. The platform is now implemented in PyTorch. With a new, more modular design, Detectron2 is flexible and extensible, and able to provide fast training on single or multiple GPU servers. Detectron2 includes high-quality implementations of state-of-the-art object detection algorithms, including DensePose, panoptic feature pyramid networks, and numerous variants of the pioneering Mask R-CNNmodel family also developed by FAIR. Its extensible design makes it easy to implement cutting-edge research projects without having to fork the entire codebase.

- **Modular, extensible design:** In Detectron2, we've introduced a modular design that allows users to plug custom module implementations into almost any part of an object detection system. This means that many new research projects can be written in hundreds of lines of code with a clean separation between the core Detectron2 library and the novel research implementation. We continue to refine the modular, extensible design by implementing new models and discovering new ways in which we can make Detectron2 more flexible.
- **New models and features:** Detectron2 includes all the models that were available in the original Detectron,such as Faster R-CNN, Mask R-CNN, RetinaNet, and DensePose. It also features several new models, including Cascade R-CNN, Panoptic FPN, and TensorMask, and we will continue to add more algorithms. We've also added features such as synchronous Batch Norm and support for new datasets like LVIS.
- **New tasks:** Detectron2 supports a range of tasks related to object detection. Like the original Detectron, it supports object detection with boxes and instance segmentation masks, as well as human pose prediction. Beyond that, Detectron2 adds support for semantic segmentation and panoptic segmentation, a task that combines both semantic and instance segmentation.
- **Implementation quality:** Rewriting Detectron2 from the ground up allowed us to revisit low-level design decisions and address several implementation issues in the original Detectron.
- **Speed and scalability:** By moving the entire training pipeline to GPU, we were able to make Detectron2 faster than the original Detectron for a variety of standard models. Additionally, distributing training to multiple GPU servers is now easy, making it much simpler to scale training to very large datasets.
- **Detectron2go:** Facebook AI's computer vision engineers have implemented an additional software layer, Detectron2go, to make it easier to deploy advanced new models to production. These features include standard training workflows with in-house datasets, network quantization, and model conversion to optimized formats for cloud and mobile deployment.

New research starts with understanding, reproducing, and verifying previous results in the literature. With Detectron2, we aim to provide high-quality reference implementations for many state-of-the-art algorithms in order to democratize this phase of the research process. The library's modular design also enables researchersto implement new projects with clean separation from standard detection library functionality. As an example,

Mesh R-CNN, FAIR's recent work on predicting per-object instance 3D meshes from 2D images, was developed in Detectron2. Detectron2's modular design enabled the researchers to easily extend Mask R-CNNto work with complex data structures representing 3D meshes, integrate new datasets, and design novel evaluation metrics. Detectron2 can be easily shared between research-first use cases and production-oriented use cases. Because the library is built in PyTorch, new models can be implemented rapidly and thentransferred to production.

### PYTORCH

The original Detectron was implemented in Caffe2. PyTorch provides a more intuitive imperative programming model that allows researchers and practitioners to iterate more rapidly on model design and experiments. Because we've rewritten Detectron2 from scratch in PyTorch, users can now benefit from PyTorch's approach to deep learning as well as the large and active community that continually improvesPyTorch. PyTorch is a machine learning library based on the Torch library used for applications such as computer vision and natural language processing, originally developed by Meta AI and now part of the Linux Foundation umbrella. It is recognized as one of the two most popular machine learning libraries alongside TensorFlow, offering free and open-source software released under the modified BSD

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Refereed Journal |

|| Volume 13, Issue 5, May 2024 ||

| DOI:10.15680/IJIRSET.2024.1305321 |

license. Although the Python interface is more polished and the primary focus of development, PyTorch also has a C++ interface.

A number of pieces of deep learning software are built on top of PyTorch, including Tesla Autopilot, Uber'sPyro, Hugging Face's Transformers, PyTorch Lightning, and Catalyst.

PyTorch provides two high-level features

- Tensor computing (like NumPy) with strong acceleration via graphics processing units (GPU)
- Deep neural networks built on a tape-based automatic differentiation system

PyTorch defines a module called nn (torch.nn) to describe neural networks and to support training. This module offers a comprehensive collection of building blocks for neural networks, including various layers and activation functions, enabling the construction of complex models. Networks are built by inheriting from the torch.nn moduleand defining the sequence of operations in the forward() function.

**FLASK**

Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools. Flask's success created a lot of additional work in issue tickets and pull requests. Armin eventually created The Pallets Projects collection of open source code libraries after he had been managing Flask under his own GitHub account for several years. The Pallets Project now serves as the community-driven organization that handles Flask and otherrelated Python libraries such as Lektor, Jinja and several others.

**YOLO**

You Only Look Once (YOLO) proposes using an end-to-end neural network that makes predictions of bounding boxes and class probabilities all at once. It differs from the approach taken by previous object detection algorithms, which repurposed classifiers to perform detection. Following a fundamentally different approach to object detection, YOLO achieved state-of-the-art results, beating other real-time object detection algorithms by a large margin.While algorithms like Faster RCNN work by detecting possible regions of interest using the Region Proposal Network and then performing recognition on those regions separately, YOLO performs all of its predictions with the help of a single fully connected layer.Methods that use Region Proposal Networks perform multiple iterations for the same image, while YOLO gets away with a single iteration.

**ULTRALYTICS**

Ultralytics is the home for cutting-edge, state-of-the-art computer vision models for tasks like image classification, object detection, image segmentation, and pose estimation. Not only it hosts YOLOv8, the latestiteration in the YOLO series of real-time object detection models, but other powerful computer vision models such as SAM (Segment Anything Model), RT-DETR, YOLO-NAS, etc. Besides providing implementations of these models, Ultralytics also provides us with out-of-the-box workflows for training, fine-tuning, and applying these models using an easy-to-use API.

**OPENCV**

Initially developed by Intel, OpenCV (Open source Computer Vision) is a free cross-platform computer vision library for real-time image processing. The OpenCV software has become a de-facto standard tool for all things related to Computer Vision. Today, OpenCV is still highly popular, with over 29’000 downloads every week. The goal of OpenCV is to provide an easy-to-use computer vision infrastructure that helps people build sophisticated vision applications quickly by providing over 500 functions that span many areas in vision. OpenCV is often used in factory product inspection, medical imaging, security analysis, human-machine interface, camera calibration, stereo vision(3Dvision) and robotic vision

**TENSORFLOW**

**TensorFlow** is a free and open-source software library for machine learning and artificial intelligence. It can be used across a range of tasks but has a particular focus on training and inference of deep neural networks. It was developed by the Google Brain team for Google's internal use in research and production. The initial version was released underthe Apache

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Refereed Journal |

|| Volume 13, Issue 5, May 2024 ||

| DOI:10.15680/IJIRSET.2024.1305321 |

License 2.0 in 2015. Google released an updated version, TensorFlow 2.0, in September 2019. TensorFlow can be used in a wide variety of programming languages, including Python, JavaScript, C++, and Java, facilitating its use in a range of applications in many sectors.

**PILLOW**

I. Pillow is the friendly PIL fork by Jeffrey A. Clark and contributors. PIL is the Python Imaging Library by Fredrik Lundh and contributors. The Python Imaging Library adds image processing capabilities to your Python interpreter.This library provides extensive file format support, an efficient internal representation, and fairly powerful image processing capabilities.The core image library is designed for fast access to data stored in a few basic pixel formats. It should provide a solid foundation for a general image processing tool. PIL, are the original Python libraries for dealing with images. Even though there are other Python libraries for image processing, Pillow remains an important tool for understanding and dealing with images.To manipulate and process images,Pillow provides tools that are similar to ones found in image processing software such as Photoshop. Some of the more modern Python image processing libraries are built on top of Pillow and often provide more advancedfunctionality.

## IV. EXPERIMENTAL RESULTS

Figure (a), (b) shows the original images

Fig (a)

Fig (b)

Figure (c), (d) image detected within the bounding box coordinates with identification numbers

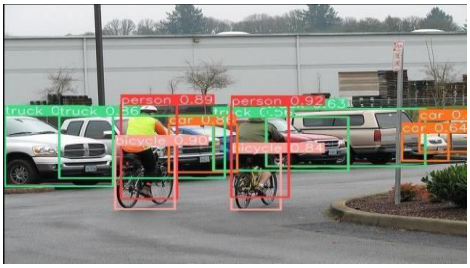

Fig (c)

Fig (d)

## V. CONCLUSION

In conclusion, multiple object detection stands as a pivotal technology with vast applications across numerous domains. Its ability to accurately identify and localize multiple objects within images or video streams has revolutionized various industries, from autonomous driving and surveillance to healthcare and retail. Through the advancements in deep learning frameworks like Detectron 2 and innovative techniques such as Faster R- CNN, YOLO, and Mask R-CNN, multiple object detection systems have become more robust, efficient, and adaptable to complex scenarios.

Despite its achievements, challenges such as background interference, illumination variations, and occlusions persist, highlighting the need for continuous research and development efforts. However, with strategies such as adaptive feature learning, multi-scale feature fusion, and contextual reasoning, these challenges can be addressed,leading to more reliable and accurate detection systems.

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Refereed Journal |

|| Volume 13, Issue 5, May 2024 ||

| DOI:10.15680/IJIRSET.2024.1305321 |

Moving forward, the integration of multiple object detection systems with other emerging technologies like augmented reality, edge computing, and the Internet of Things (IoT) holds immense potential for further innovation and transformative applications. As the field continues to evolve, collaboration among researchers, developers, and stakeholders will be crucial in pushing the boundaries of multiple object detection and unlocking its full potential for the benefit of society.

Research and implement efficient training techniques such as mixed precision training, gradient checkpointing, ormodel distillation to reduce memory consumption and training time without sacrificing performance. Explore advanced data augmentation techniques tailored for object detection tasks. This could include domain-specific augmentation strategies or techniques inspired by recent advancements in data augmentation research.

## REFERENCES

1. "Release 3.0.3". 7 April 2024. Retrieved 23 April 2024.
2. "Flask Foreword". Archived from the original on 2017-11-17. [3]^ "Flask Extensions". Archived from the original on 2018-05-17. [4]^ What challenges has Pinterest encountered with Flask? [5]neering and Technology, vol. 1 Issue 2, ISSN 2320 – 4486, 2013.
3. Uday Modha, Preeti Dave, " Image Inpainting-Automatic Detection and Removal of Text From Images", International Journal of Engineering Research and Applications (IJERA), ISSN: 2248-9622 Vol. 2, Issue 2, 2012 Muthukumar S, Dr.Krishnan .N, Pasupathi.P, Deepa. S, "Analysis of Image Inpainting Techniques with Exemplar, Poisson, Successive Elimination and 8 Pixel Neighborhood Methods", International Journal of Computer Applications (0975 – 8887), Volume 9, No.11, 2010 https://www.analyticsvidhya.com/blog/2021/08/your-guide-to-object-detection-with-detectron2-inpytorch/ https://stackoverflow.com/questions/67061435/how-to-train-detectron2-model-with-multiple-ustomdataset https://medium.com/@mumbaiyachori/training-detectron2-object-detection-custom-dataset-with-ease-cdcdadfe5842
4. https://pjreddie.com/darknet/yolo/[12]https://www.tensorflow.org/ [13]https://pypi.org/project/pillow/

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi cross ref

निस्केयर NISCAIR

# INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

## IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com

www.ijirset.com

Scan to save the contact details