3-Dimensional Object Detection Using Deep Learning Techniques

S. Bharathi, Piyush Kumar Pareek, B. R. Shobha Rani, and D. R. Chaitra

Abstract Computer Vision is one of the branches of computer science. It will detect and understand the images and scenes. This work suggests a real-time, immediate motion tracking system for devices that follows an object's attitude in space as represented by its 3D bounding box. Computer Vision includes different features such as image recognition, image production, object detection, high-resolution image processing etc.… Object detection is frequently utilized in self-driving cars, security systems, facial recognition, pedestrian counts, and online photos. The most accurate acquisition algorithms and techniques are used in this research. This covers the precision of each identifying technique. Images can contain objects that can be automatically located and recognized. One of the core issues with computer vision is object detection. This work will show that the most cutting-edge approach to object detection at the moment is R-convolutional neural networks. This is the major objective is to examine and evaluate convolutional object identification techniques. The main idea behind such system is to classify various images and to classify object's position approximately in all the images in order to give a full information about the images and videos. The system will be able to detect, localize and classify several objects using given image or videos. It is very difficult to classify images into different classes.

Keywords 3D object detection · R-CNN · MediaPipe · TensorFlow · OpenCV

S. Bharathi (✉)
Department of MCA, Dr. Ambedkar Institute of Technology, Bengaluru 560056, India
e-mail: bharathishivu2020@gmail.com

P. K. Pareek
Department of AI & ML, Nitte Meenakshi Institute of Technology, Bengaluru, India

B. R. S. Rani · D. R. Chaitra
Dr. Ambedkar Institute of Technology, Bengaluru, India

N. R. Shetty et al. (eds.), *Advances in Computing and Information*, Lecture Notes in Electrical Engineering 1104, https://doi.org/10.1007/978-981-99-7622-5_21

1 Introduction

Detecting object is a technique connecting with the problem of object detection from photos, or videos that fall under a particular category is the main aim of computer vision and image processing. Object detection in 3D image plays an important role in designing effective real-world systems for recognizing and detecting all recognised objects in a image is the main goal of object detection. There are various purpose of locating and recognition of the objects includes counting the object, face recognition, character recognition, and independent driving in a surveillance camera.

Many teams of engineers and scientists are working to address each of these "eternal" challenges through trial and error. As modern technology solutions seem to be more expensive, the task of creating self-sufficient software tools for problem solving is being developed and solved in depth overseas.

A vast number of 3D objects are frequently utilized in 3D graphics, which is becoming more popular throughout the world. In order to use 3D objects, an extraction method is required. This work explains how to visualize a 3D object using the deep learning technology. There has been a lot of study on object detection, but most of it has focused on identifying objects from 2D pixel data generated from camera. In this instance, objects are identified by referring pixel data.

Objects in the image can be seen and pointed out by viewers. The human visual system is very accurate and rapid, it can do challenging tasks including instantly differentiating between a variety of obstacles. Now computers probably learn to recognise and differentiate various components inside an image, due to the availability of larger data sets, faster GPUs, and smarter algorithms.

While the local rendering of an object is painting a small rectangular or square box around objects in a photograph, image classification entails classifying an image. These tasks, drawing a rectangular or square box around each and every object of interest in the image, makes obtaining an object always a difficult task. All of these topics are cited as object recognition issues.

Identifying objects in digital photos involves a number of related processes known as object recognition. In order to deal with local practise and monitoring operations, a family of methods known as regional-based convolutional neural networks or R-CNNs is used to develop a model.

2 Related Work

Hough space of LiDAR point clouds is used to overcome the problem arises such as unstructured distribution, disordered arrangement, and large amounts of data. This requires high computational complexity and it is very difficult to classify 3D objects. CNN model is used to classify multidimensional objects [1].

Perceptual organization technique is used configure the grouping and structure which is unchangeable over a wide range of viewpoints to reduce the size of the search space during model-based matching, probabilistic ranking technique is used. The last technique used is spatial correspondence brings the projections of three-dimensional models into direct correspondence with the image by solving for unknown viewpoint and model parameters [2].

Ning Hao discuss the 3D object detection using three stage objection model. This model improves the efficiency compared with traditional 3D object detection model. Point CNN model for 3D object detection is also discussed [3]. A F M Saifuddin Saif et al. discuss the models, challenges and applications using deep learning techniques for multimodal object detection in future [4].

Current techniques for object recognition uses efficient machine learning methods. To optimize the performance, huge datasets are used, apply more powerful models, and techniques to prevent overfitting. ImageNet classification with deep convolutional neural networks is an efficient technique to classify high resolution images. The neural network, which has 60 million parameters and 650,000 neurons, consists of five convolutional layers, some of which are followed by max-pooling layers, and three fully connected layers with a final 1000-way SoftMax [5].

Deep learning techniques are widely used in 3 D object detection in the field of automated driving, medical image analysis, virtual/augmented reality, artificial intelligence robots, and other areas. This is a very challenging and is a hot research topic in the current scenario. There are many techniques existing to solve the problems, but deep learning methods gives optimized result [6].

3 Methodology

3.1 Objectives

- Identifying 3D objects from an image.
- Classifying and assign the labels to each object.

3.2 Methodology

Traditionally object recognition was carried out using hand-crafted features like HoG or Haar before the advent of contemporary Convolutional Neural Network (CNN) architectures and large-scale picture datasets like ImageNet. CNNs were first used to recognize objects using externally segmented objects, and later, Region Proposal Networks (RPNs) were used.

Three methods have been used to detect 3D objects. The first strategy, (among many others) relies entirely on monocular visual information to estimate the spatial placement of the objects. A second strategy has attempted to combine the data, as in using both the camera and LiDAR as complimentary data sources. RPN was utilized in this data fusion approach to compute regions of interest and classification in the picture space and perform final location over the LiDAR data.

The third method of 3D object detection computes object detections in 3D using point cloud data and either information from stereo cameras or LiDARs. Examples of this strategy from the past include and more recently, 3D space has been transformed into a voxel grid and 3D convolutions have been used to analyze the spatial data from point clouds. In the point cloud category, using 2D CNNs on a LiDAR point cloud of the front view or a bird's eye view (BEV) is a relatively recent innovation.

To enhance the problems of existing system and to make it better for the Object detection and identification proposed system has been implemented. Object detection Technology is used to identify the object by using python Models and libraries like TensorFlow and OpenCV.

In this work, several commonly used datasets for 3D object detection are reviewed and corelated them with Objectron. The computer vision problem of object detection has been extensively investigated. However, 2D object prediction has received the majority of attention. A various application in robotics and automation, self-driving vehicles, image capturing, and augmented reality are made possible by improving prediction to 3D, which allows one to record the size of object's location and object orientation in the real world. 2D prediction only offers 2D bounding boxes. Even though 2D machine vision is very advanced and has been broadly used in the industries, 3D object detection from 2D photography is a challenging challenge to address due to the insufficient data and the range of data appearance and shapes of items within a category.

Image preprocessing. Pre-processing has been done to enhance the image quality, then image analysis is more effective and successful. Through preprocessing, undesirable distortions are eliminated and enhance certain properties of the image that are important for the application. Those traits are altered depending on the application. During preprocessing image data is transformed into the format compatible for algorithm to process and data analysis.

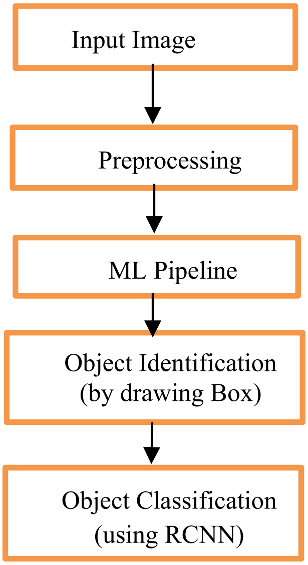
R-CNN. The R-CNN algorithm is mainly used for progressive visual object detection which fuses convolution neural network generated replacement with bottom-up region approach. R-CNN uses a technique called region proposal to generate prospective bounding boxes around the images before implementing a classifier to the recommended boxes (Fig. 1).

System architecture

MediaPipe. Machine learning pipelines are built using the MediaPipe framework to handle real time including audio and video. This cross-platform system is supported by the desktop/server, Android, iOS, and embedded systems like the Raspberry Pi and Jetson Nano.

Because of the ease of setup and ubiquity of the Python programming language, MediaPipe Python solutions are the best for beginners. The MediaPipe Framework's flexibility allows for customization. But before diving into customizing, we advise being familiar with a number of ready-made solutions. Recognize the internal APIs connected to them, then modify the results to produce your fascinating apps.

Fig. 1 System architecture


Input Image
Preprocessing
ML Pipeline
Object Identification (by drawing Box)
Object Classification (using RCNN)

ML Pipelines for 3D Object Detection. To estimate the bounding box around 3D object from a single RGB image, this system developed two pipelines: a two-stage pipeline and single-stage pipeline. While maintaining and improving efficiency, the two-stage pipeline is three times faster than the one-stage pipeline. The two-stage pipeline excels in detecting a single dominating item, the single-stage pipeline excels in detecting multiple objects.

Two-stage pipe. The illustration in Fig. 2 shows our two-stage pipeline. In the first stage 2D crop of the object is located in using an object detector. The second stage calculates the 3D bounding box using the image crop. In order to avoid running the object detector every frame, it additionally computes the cropped object from 2D image for the following frame.

Single-Stage pipe

This framework uses multi-task learning strategy such as combining, detection and regression to estimate jointly the shape of an item. Depending on the ground truth, annotation is provided. This is not required, if the shape of the image is not proper for identification in the training data (Fig. 3).

Box Tracking. The box tracking system takes image frames from a video or camera stream and computes the monitored box positions for each frame by starting box positions with timestamps, signaling 2D region of interest to track. The starting box positions in this particular use case are determined by object detection, but the starting position can also be supplied directly by the user or by another system. Three key parts make up our solution: a motion analysis part, a flow packager part, and a box tracking part. The box tracking solution is represented as a Media Pipe subgraph, with each component represented as a Media Pipe calculator.

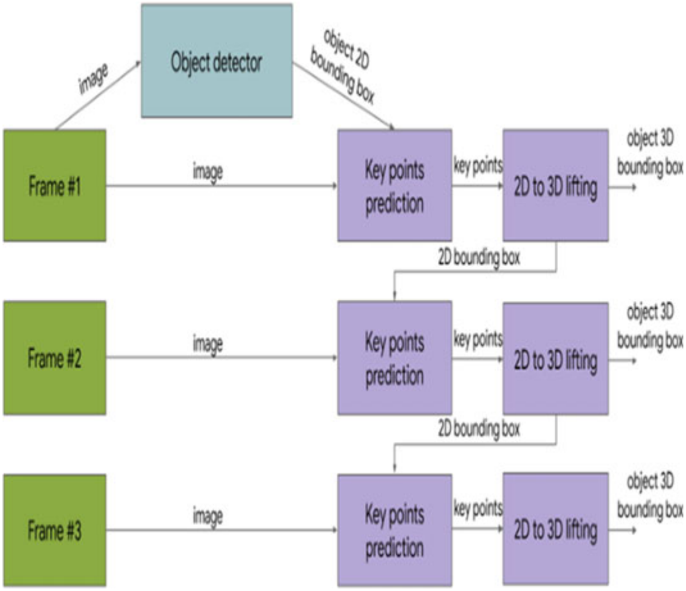


Fig. 2 Two stage pipe

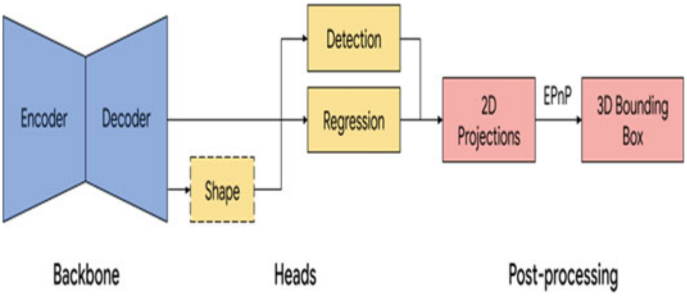


Fig. 3 Single-stage pipe

Coordinate Systems. Every item is having its own coordinate frame. The source is in the center of the 3D bounding box with +x pointing right, +y pointing up, and +z pointing forward. The object coordinate specification shown below (Fig. 4).

Camera Coordinate. Scale, rotation, and translation in relation to the camera coordinate frame are the parameters that define a 3D object. The definition of the, the camera coordinate API is as follows: +x points to the right, +y points to the up, and −z points to the scene. To work with box landmarks, one can first derive landmark coordinates in object frame by scaling a origin centered unit box with scale, then transform to camera frame by applying rotation and translation (Fig. 5).

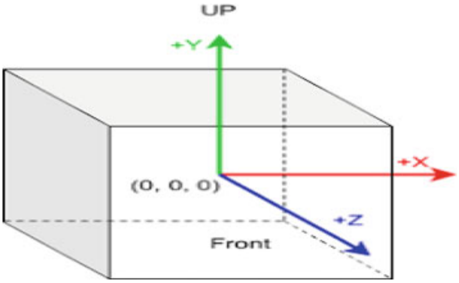


Fig. 4 Coordinate system

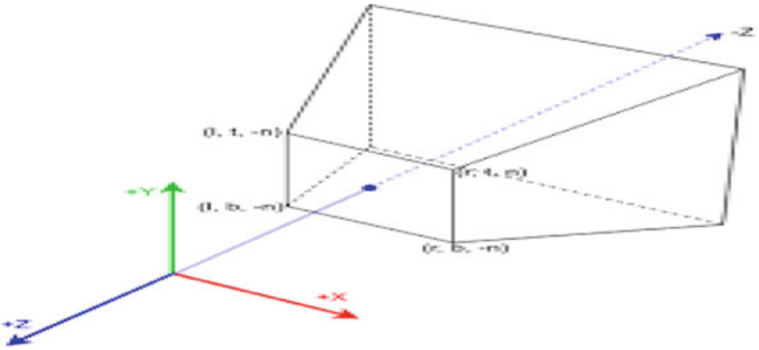


Fig. 5 Camera coordinate

TensorFlow. To locate objects, a TensorFlow object identification API based on an SSD deep learning framework was employed. We were able to use the model weights provided by this API since it has previously learnt 90 items. For this research, the detection model's source code was studied and updated in order to pinpoint the precise position of objects in real time. Using the source code, the camera's coordinates were found. It is possible to recognize and locate things in an image or video using the object detection computer vision technology. For example, object detection may be used to count and monitor the exact positions of all the items in a scene while precisely identifying each one of them.

OpenCV. The box tracking system takes image frames from a video or camera stream and computes the monitored box positions for each frame by starting box positions with timestamps, signaling 2D regions of interest to track. The starting box positions in this particular use case are determined by object detection, but the starting position can also be supplied directly by the user or by another system. Three key parts make up our solution: a motion analysis part, a flow packager part, and a box tracking part. The box tracking solution is represented as a MediaPipe subgraph, with each component represented as a MediaPipe calculator.

4 Results and Performance Evaluation

Initially the system will capture the input image from webcam after that images will pre-processing in order to get an enhanced image and to extract some useful information and it improves the image quality. 3D object uses the pixel values at that point of an image and describe how bright that pixels, and what color it should be. The computer reads any image as a range of values between 0 and 255 using RGB colors it will plot the pixels. For feature extraction a MediaPipe multistage pipeline is applied. This is known as MediaPipe Holistic. The input taken from web camera, the MediaPipe Holistic uses individual models for each object components using a region-appropriate image resolution.

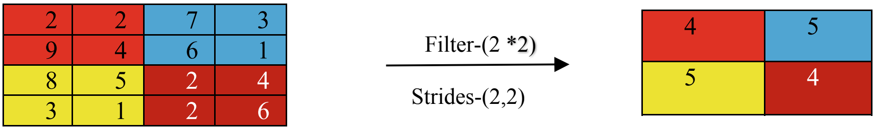
Next R-CNN algorithm will apply. The region of interest or region proposal will be generated by R-CNN in the I stage. Selective search algorithm will be use to segment the image. This algorithm will divide the image based on size, texture, color and shape. At this stage object proposal will be generating by R-CNN pipeline, it is of different scale. This object proposal is challenging to region proposal. Different features have to extract from the proposal. SVM classification algorithm will be used for classifying and labeling the images. To improve the performance of prediction bounding box regression has been applied.

The main goal is to detect objects in any input image and drawing boundary around them. This technique is used to create potential bounding boxes for images before applying to the suggested box after applying the R-CNN MediaPipe objectrone. This MediaPipe objectrone is a real-time 3D object detection solution for every objects. It detects objects in 2D images and also estimate their poses. LiDAR dataset is used as a training dataset. The accuracy obtained is better and more accurate after using this technique. This proposed work is trained and undergone 300 epoch and achieved 90 percent accuracy.

Table 1 Algorithm

Input: Training dataset
Output**:** To create data model for 3D object detection

Step 1: Installing and import dependencies
Step 2**:** To determine the object, crop the input image, the first stage used the popular TensorFlow object identification model
Step 3**:** The second step require these cropped photos to determine their 3D bounding boxes
Step 4: Performing object detection and classification by applying RCNN technique
Step 5: Train the model, validation accuracy and show the loss
Step 5: Draw the box around the image and assign the class



2	2	7	3
9	4	6	1
8	5	2	4
3	1	2	6

4	5
5	4

Fig. 6 Average pooling feature map

4.1 Pooling Layers

Pooling is a technique in CNN for generalizing feature extraction by convolutional filters and helping the neural network to recognize features independent of their location in the image. This method gives the average of features present in the image. The below table figure shows the feature map of average pooling.

Max pool (Figs. 6, 7, 8, 9, 10, and 11).

Fig. 7 3D bounding object detection chairs

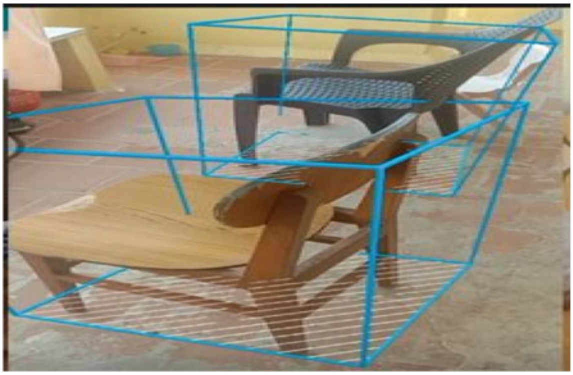
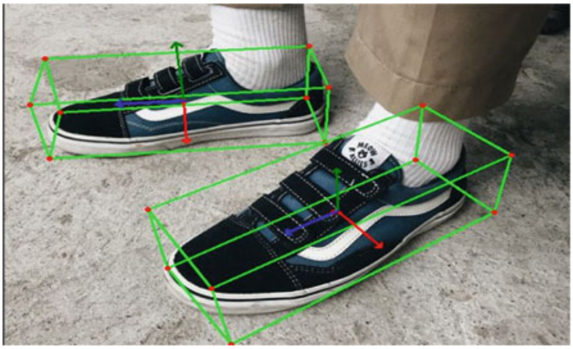


Fig. 8 3D bounding object detection shoes



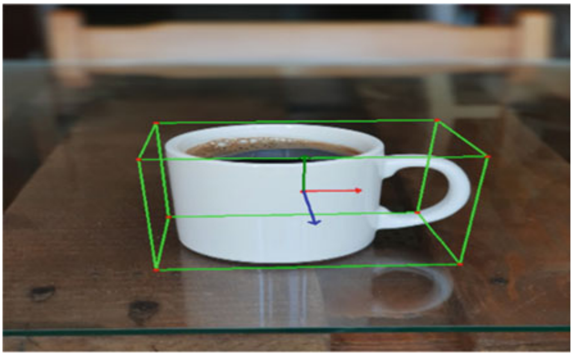


Fig. 9 3D bounding object detection cup

Fig. 10 Original image

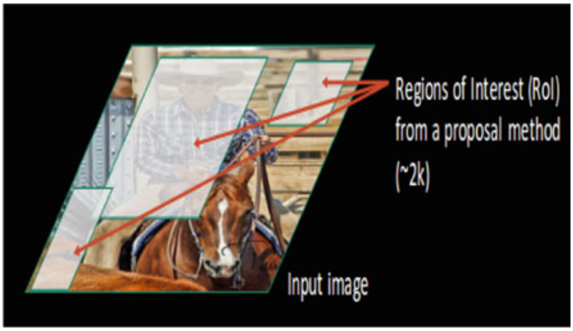


Fig. 11 ROI using selective search method

5 Conclusion

This framework provides a tracking system for 3D objects that enables immediate real-time tracking of 3D bounding boxes on devices. This system recommends a technique which initializes the 3D posture first with a neural network, then uses a planar surface tracker to follow the object's pose through all the images.

The main goal of the System is "3D Object Detection System " is to identify several items in various kinds of images especially in medical images. In order to accomplish this, edge features from the image are retrieved. It makes use of a vast image database to accurately identify and recognize objects. The accuracy obtained from this method is 90%. The user interface for this system will make it simple to retrieve the desired photographs. In the Future research work developing a robust model for multidimensional object detection and classification in real time images.

References

1. Song W, Zhang L, Tian Y, Fong S, Liu J (2020) Gozho A (2020) CNN-based 3D object classification using Hough space of LiDAR point clouds. Hum Cent Comput Inf Sci 10:19. https://doi.org/10.1186/s13673-020-00228-8
2. Lowe DG (1987) Three-dimensional object recognition from single two-dimensional images. J Artif Intell 31(3):355–395
3. Hao N (2022) 3D object detection from point cloud based on deep learning. 2022:6228797. https://doi.org/10.1155/2022/6228797
4. Saifuddin Saif AFM, Rasyid Mahayuddin Z (2022) Vision based 3D object detection using deep learning: methods with challenges and applications towards future directions (IJACSA). Int J Adv Comput Sci Applicat 13:11
5. Krizhevsky A, Sutskever I, Hinton G (2012) IMAGENET classification with deep convolutional neural networks. Adv Neural Inform Process Syst 25(2)
6. Shaohua Q, Xin Ning G, Yang L, Zhang Peng L, Weiwei C, Weijun L (2012) Review of Multiview 3d object recognition methods based on deep learning. Elseiver. vol 69