Computer Systems Science & Engineering
DOI:10.32604/csse.2022.022590
Article

Tech Science Press

A Fast Panoptic Segmentation Network for Self-Driving Scene Understanding

Abdul Majid[1], Sumaira Kausar[1,*], Samabia Tehsin[1] and Amina Jameel[2]

[1]Department of Computer Science, Bahiria University, Islamabad, Pakistan
[2]Department of Software Engineering, Bahiria University, Karachi, Pakistan
*Corresponding Author: Sumaira Kausar. Email: sumairakausar.buic@bahria.edu.pk
Received: 12 August 2021; Accepted: 28 October 2021

Abstract: In recent years, a gain in popularity and significance of science understanding has been observed due to the high paced progress in computer vision techniques and technologies. The primary focus of computer vision based scene understanding is to label each and every pixel in an image as the category of the object it belongs to. So it is required to combine segmentation and detection in a single framework. Recently many successful computer vision methods has been developed to aid scene understanding for a variety of real world application. Scene understanding systems typically involves detection and segmentation of different natural and manmade things. A lot of research has been performed in recent years, mostly with a focus on things (a well-defined objects that has shape, orientations and size) with a less focus on stuff classes (amorphous regions that are unclear and lack a shape, size or other characteristics Stuff region describes many aspects of scene, like type, situation, environment of scene etc. and hence can be very helpful in scene understanding. Existing methods for scene understanding still have to cover a challenging path to cope up with the challenges of computational time, accuracy and robustness for varying level of scene complexity. A robust scene understanding method has to effectively deal with imbalanced distribution of classes, overlapping objects, fuzzy object boundaries and poorly localized objects. The proposed method presents Panoptic Segmentation on Cityscapes Dataset. Mobilenet-V2 is used as a backbone for feature extraction that is pre-trained on ImageNet. MobileNet-V2 with state-of-art encoder-decoder architecture of DeepLabV3+ with some customization and optimization is employed Atrous convolution along with Spatial Pyramid Pooling are also utilized in the proposed method to make it more accurate and robust. Very promising and encouraging results have been achieved that indicates the potential of the proposed method for robust scene understanding in a fast and reliable way.

Keywords: Panoptic segmentation; instance segmentation; semantic segmentation; deep learning; computer vision; scene understanding; autonomous applications; atrous convolution

1 Introduction

Computer vision is at the core of many high tech systems in the field of medical, robotics, entertainment, industry, education etc. Many researchers are working on recognition of objects and action using brain-inspired algorithms which use various classifiers and feature detectors with the help of machine learning. These techniques are called deep learning techniques which have shown outclass results on different challenging datasets. Segmentation is one of the biggest challenge in computer vision system especially when it comes to scene understanding. With the steady progress in computer vision techniques, different types of segmentations has been introduced e.g., object segmentation, semantic segmentation, instance segmentation etc. In semantic segmentation task is to label each pixel of an image with a class label. The Task of instance segmentation is same as Semantic Segmentation, but dives a bit deeper [1], it detects each pixel and separates the instances from one another. It is intended to propose a task which includes both semantic and instance segmentation for videos, by using a single-network called baseline for the joint task termed as panoptic segmentation. Hence, semantic segmentation is widely a task of per-pixel predictions of semantic labels. Whereas instance segmentation simply includes detection of the objects at the instance level.

Video is a temporal sequence of static images. It gives track of continuous motion when taking into consideration repeatedly and rapidly. Video comprises surplus information, its adjacent frames carry related information. Mostly video sequences are segmented by processing all the frame of the video separately [2]. In each frame of video sequences a label is assigned to each pixel from predefined object classes like cycle, car, buses, persons, animals, water and building etc. Independent processing of each frame would allow parallelization the computation. It would not be impacted by sequence interruptions to process the frames at any desired rate.

Semantic segmentation in dense nature would entail a cost intensive dataset annotation process. Annotation is one of the most eminent tasks in computer vision. Quality of annotation has a vital role in training a better model. Whereas coarse or noisy annotations are substantially cheaper to collect as annotating large adjacent regions can be achieved speedily using annotation tool kits but these can lead to poor scene understanding. Collection of large amounts of finely annotated data along with object boundaries is highly challenging and expensive. It mostly involves unclear pixels having fuzzy boundaries (Fig. 1) [3].

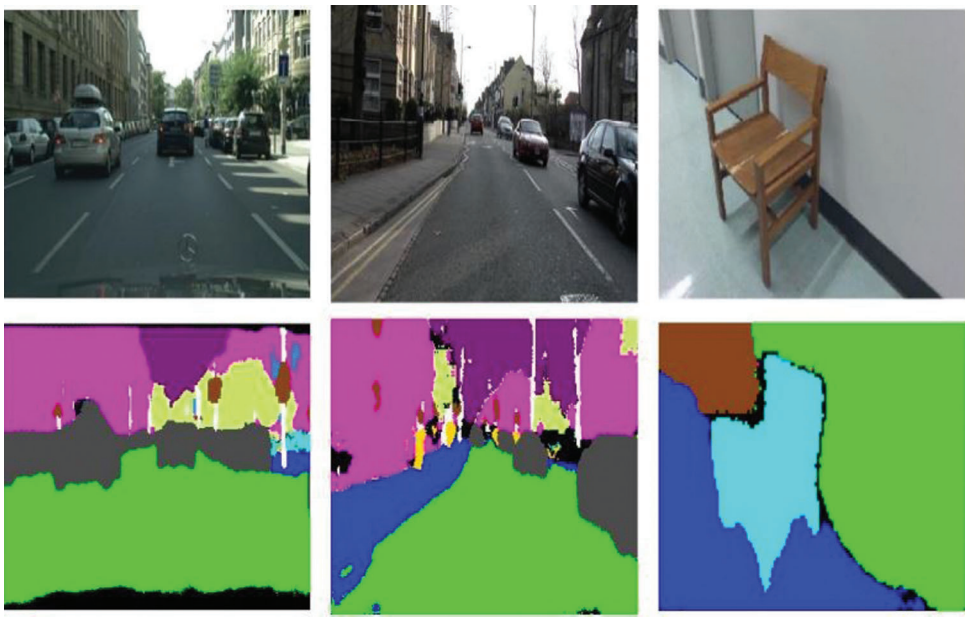

Figure 1: Fuzzy boundaries

2 Segmentation

Computer vision is usually implementation of machine learning combined with image processing techniques to give machines, human like vision capability. Computer vision has wide application areas such as agriculture, healthcare, education, entertainment, robotics, surveillance, transportation, manufacturing and many more. Human being has outstanding vision capabilities that perceive and understand their visual surroundings very quickly and very precisely even with incomplete and vague visual information. This concept of unconscious inference is closely related to visual perceptual psychological theory of humans [4], that human brain builds assumptions and conclusions from incomplete data, based on previous experiences. But training a computer to build the artificial vision system is extremely challenging. Scene understating is a pixel-level segmentation task, each and every pixel needs to be considered for proper understanding. Due to limited training data and resources, it's not easy to build an accurate model.

Humans visually interpret the world as a set of entities, like (truck, person, cat) along with their features including color, shape, position, orientation etc. and also focus on background materials like (roads, walls, water) which has no proper shape or size. Some of the background stuffs are not too much clear but humans build their visual understanding about scene using clear things or some prior information. Some contextual clues also help to build perception like car is usually found on road or ship is not mostly found on ground. So, these clues are our prior information that's very helpful in scene understanding. According to the Gestalt psychology stuff (background uncountable objects) and things (foreground objects that are countable) are closely related to perceptual grouping. One of the similar concepts is about mass and count distinction by [4–6], that human has more focus on countable things because most things are man-made as compared to stuff or background objects.

In early 1970s research conducted in computer vision, was more focused and gained lot of attention on primitive shape detection (things objects having proper shape orientation) along with different templates matching scheme to achieve more accuracy Because the focus is on man made things (things- items), it is very difficult for such methods to generalize the real-world images. In 2000s, significant success was achieved by researchers in detection of face and detection of pedestrians. Now in the detection of thing classes, spatial information of different parts plays vital role to get the right prediction. Many researchers are working on the recognition or detection of realistic images, in researchers achieved significant breakthroughs to generalize the realistic images and perform training using dozens of thing classes for object detection and also benchmark their result.

However on the other hand stuff class could not get due attention from the researchers. Some texture classifications were focused earlier Recent researches have been done with real-world images but semantics were not considered in it. In 2015 researcher filled the gap of semantic segmentation by performing classification of each pixel in an image using stuff and thing classes. In early datasets of semantic segmentation contains stuff and thing annotations. Unfortunately, with the passage of time dataset changed and the most popular dataset PASCAL VOC 2012 covered thing classes [7]. The progress in image segmentation techniques is summarized in Fig. 2.

3 Segmentation Types

Basically, segmentation is a process in which image is divided into multiple segments. These segments are simpler representation of image that further contribute in image analysis. Segments are based on different features (like patterns, color and size). A well-defined segment of image is different from all the other segments of image. There are too many ways to create a segment. Segmentation is a pixel level task where each and every pixel is labeled. For the better scene understanding it is very important to segment all the pixels that uniquely identifies all the things present in a scene. There are many types of

segmentation, but in context of scene understanding, one possible categorization can be: semantic segmentation, instance segmentation and panoptic segmentation. In panoptic segmentation the objective is to segment all the pixels in an image for all the predefined classes and to identify all the instances present in an image. Panoptic segmentation performs joint collaboration of both the tasks of semantic and instance segmentation. In semantic segmentation each and every pixel is labeled. Semantic covers both the stuff and things in image, but it could not recognize any difference between stuff and things and their overlapping areas [2]. Semantic segmentation considers all the objects of same class as a single stuff region and it does not uniquely identify individual instances. For the recognition of individual instances, object detection along with identification of every pixel (localization) it belongs to can be performed. It is known as instance segmentation. So panoptic segmentation contains semantic and instance labels for every pixel in an image. If the pixel is identified in a stuff region, then it will assign a dummy value otherwise each pixel has its instance label that represents a specific object pixel. In panoptic segmentation each and every pixel is labeled along with which instance classification. Panoptic segmentation is a new task in computer vision specially to understand the scene densely.

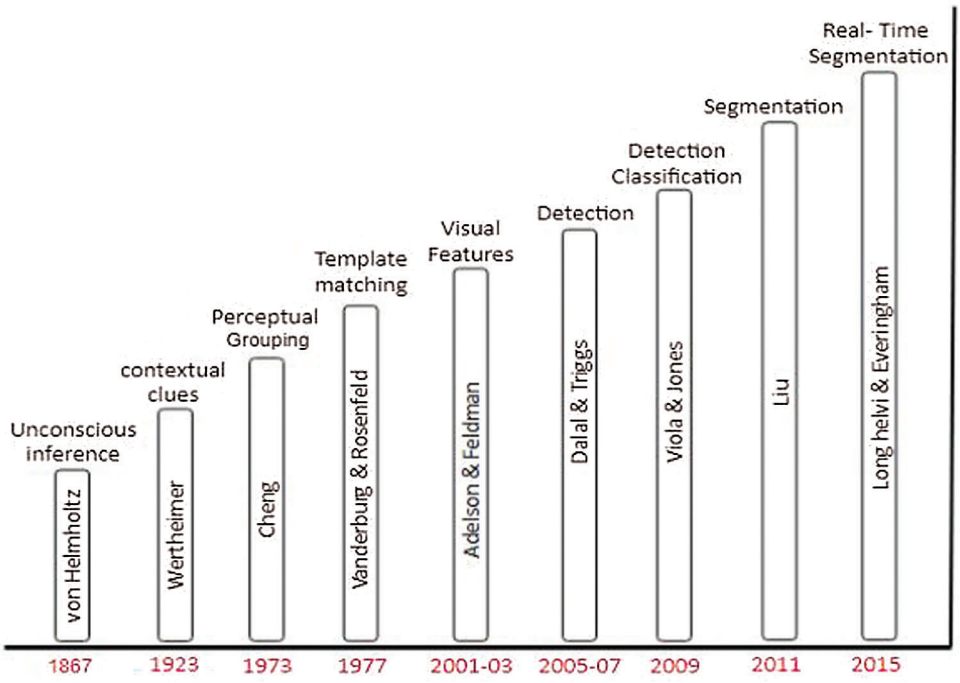

Figure 2: Segmentation

Difference between different types of segmentation is shown in Fig. 3.

3.1 Semantic Segmentation

In semantic segmentation each and every pixel is classified into a set of predefined semantic classes. Classes should be a stuff (roads, sky, trees, etc.) or things (any countable objects having a specific shape, size, orientation like, person, car, cycle, etc.). This task is pixel level task. So, output of semantic segmentation is that each and every pixel in image is assigned with a class label. Semantic segmentation shows where and what the different classes are present in image. In this approach of pixel-level classification, objects of the same class are grouped together [8]. Due to this it will differentiate between different classes but not any individual instances of same class.

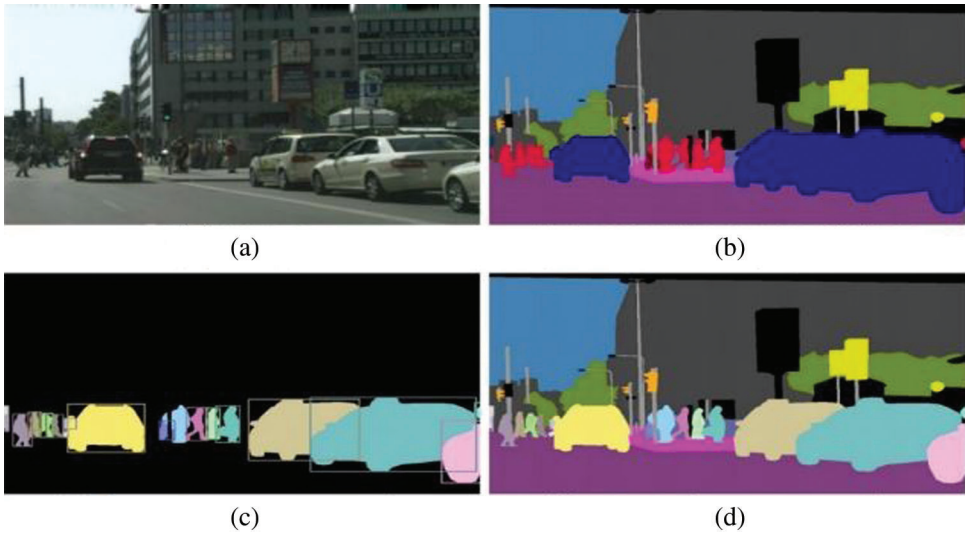

Figure 3: Difference between Segmentation and its types. (a) Image (b) Semantic segmentation (c) Instance segmentation (d) Panoptic segmentation

There are two main mechanisms that are used to perform semantic segmentation. All the published architectures [9–20] of semantic segmentation works on the basic principle of region based mechanism or full-convolutional based segmentation.

3.2 Instance Segmentation

Instance segmentation is also a pixel-level task, which locate the objects in the image by assigning specific class to each object and generating a pixel-based mask for it. It is more challenging than semantic segmentation. Basically, in this task object detection is involved with localization. In contrast to semantic segmentation this approach segments each instance individually (Fig. 4).

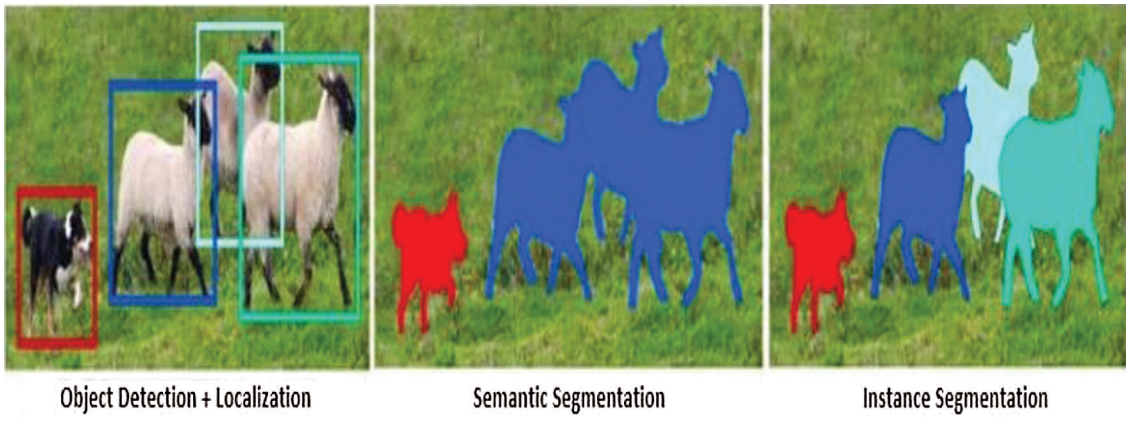

Figure 4: Instance *vs*. Semantic Segmentation

Instance segmentation is more expensive but a robust segmentation. There are two main approaches to perform instance segmentation. First one is Top-Down (Proposal-Based Method) and second one is Bottom-Up (Segmentation based or Proposal-Free method).

3.3 Panoptic Segmentation

In panoptic segmentation for scene understanding we need to label all the pixel in the image according to the pre-defined classes along with all the individual objects of that class they belong to. It's an emerging and more challenging task [21–25]. Basically, in panoptic segmentation two different concepts of segmentation are used together. Semantic and instance segmentation are jointly performed in panoptic segmentation. Semantic segmentation is used to label all the pixels in an image, whether it is stuff (uncountable classes, like road, trees, sky etc.) or things (countable objects, like cars, persons, bikes etc.). Instance segmentation is used to identify things classes only, having all the individual instances of the same class are uniquely represented. Panoptic segmentation aims to label each and every pixel with two ids' (semantic-id and instance-id). So panoptic segmentation has its own evaluation criteria, due to two separate segmentation networks merged together using a heuristic post-processing. Mostly the researcher used single network architecture that includes both semantic and instance approaches together. According to the literature review there are two different segmentation mechanism occurs in a single architecture of panoptic segmentation that are: Dependent Approaches of (Semantic-Instance Segmentation) and Separate Approaches of (Semantic-Instance Segmentation).

4 Related Work

In this session, we briefly review eminent development in panoptic segmentation architectures.

In [21] an end-to-end unified network for panoptic segmentation proposed that composed of Feature pyramid network (FPN). Panoptic head has two inputs. Firstly, a feature map is used as input in which it can perform dense segmentation. Secondly an attention masks is used as an input that indicates the existence of things, instances and the classes correspond to those instances based on Retina-Net used with Resnet-50 architecture. And addressing the problem of separate semantic and instance approaches [22] they proposed a single architecture that contains the Region Proposal Network (RPN) as a backbone for single architecture to provide shared representations. For solving these tasks simultaneously there are two heads on top of the backbone. First Head is a Semantic head based upon deep residual network (Res- Net) and utilizing multi-scale information from feature pyramid networks (FPN) [21]. And Second head is instance head that based on the Mask-RCNN [22] design and outputs mask segmentation and its associated class.

FAIR (Facebook AI Research) [23] build a model that covers both instance and semantic task in a single network [23]. In this architecture MASK-RCNN is used which extends Faster-RCNN for region-based object detector. And FPN is used as a backbone with a restnet-101 pertained on image net. Instance perform on every level of FPN while semantic segmentation is performed on the deepest level of FPN [23]. Many researchers try to improve accuracy and computational problems in panoptic architecture but in [24] occlusion problems are more focused. In this study, an end-to-end Occlusion Aware Network (OA-Net) for panoptic segmentation is proposed. This single network contains FPN as a backbone network with Resnet-50. For instance segmentation, Mask-RCNN is used but using top-down path way to extract RPN feature map as an input for instance segmentation. Same RPN-feature map is stacked for the semantic segmentation. Spatial Ranking Module (SRM) [24] that overcome the problem of occlusion is proposed. Due to SRM algorithm, mapping of the objects in the feature map is performed. And the pixel-wise cross entropy loss calculation is performed for optimization of ranking scores of non-semantic overlaps. In [25] this architecture Resnet-50 is used as a features extractor for both the tasks of semantic and instance

segmentation. But for semantic segmentation only few parameters of feature extractor are used, due to this Pyramid Pooling Module is used (PPM) [25]. Mask-RCNN with RPN (Region Proposal Network). is used for instance segmentation, For joint learning of instance and semantic segmentation, RPN adds bounding boxes along with regions based on the semantic segmentation output. Then detection branch predicts the bounded boxes based on the semantic segmentation output. One of the major problems in panoptic segmentation is the overlapping prediction conflict in semantic and instance heads. Previous approaches [21–25] faces the challenges of computational efficiency, slow runtime, redundancy in learning and computational overhead due to disjoint approaches. In this architecture [26] they also proposed a novel Efficient-PS architecture that consist of an encoder with 2-way Feature Pyramid Network (FPN) followed by separable convolutions of instance and semantic heads. Semantic head consist of three different modules. First, employ Large-Scale Feature Extractor (LSFE) at large-scale to capture the fine features efficiently. Second, employ Efficient Atrous Spatial Pyramid Pooling (eASPP) at small-scale to capture long-range context. And the third and last one of the semantic head is Mismatch Correction Module (MC) to migrate the mismatch between large-scale and small-scale features regression. In instance head the researchers have [26] used MASK-RCNN. Finally, to obtain the output for panoptic segmentation, this approach used Panoptic Fusion Module (PFM) that fuse the prediction of semantic and instance head. To resolve the problems of Top Down approaches, Google Research Team provide a fast, simple and effective novel architecture [27] based on bottom-up (Proposal Free) method. Proposed novel architecture consist of an Xception-71 encoder backbone that is pertained on Image-Net used with Atrous Convolution for extraction of dense feature maps. In this approach both semantic and instance head has separate Atrous Spatial Pyramid Pooling (ASPP) and decoder modules. Decoder module that follows in this architecture [27] is DeepLabV3+, which is a light weight decoder. For panoptic segmentation they simply used grouping operation of instance and semantic segmentation using Majority Voting algorithm and compute the class-agnostic scores from semantic segmentation prediction.

5 Methodology

The proposed architecture is a bottom-up (Proposal free) method .In order to overcome the conflict of overlapping instances or duplicating pixel-wise prediction, occurs in proposal-based methods, the proposed method uses a proposal-free approach to improve the prediction. DeepLabV3+ a state-of- the-art architecture especially designed for segmentation by Google has been adopted. It consists of four major components. (1) A shared encoder backbone with Atrous convolution for both the tasks of instance and semantic segmentation to extract and preserve the dense feature information like to generate the feature map at original resolution. (2) Separate ASPP (Atrous Spatial Pyramid Pooling) module for both the semantic and instance tasks to obtain the contextual information at multi-scale. (3) Decoupled decoder modules apply on multi-scale contextual information. (4) Separate prediction head for every task. A post processing method for panoptic segmentation is also proposed. An over view of the proposed architecture is shown in Fig. 5.

5.1 Encoder-Backbone Using Atrous Convolution

In our proposed model we use MobilNetV2 as a backbone architecture that is pre-trained on the ImageNet. This architecture is used with its official weights that is being available on the pytorch official website. The encoder backbone is shared between semantic and instance task. Atrous-Convolution in encoder-backbone for extraction of the dense feature maps is used. Every pixel with its contextual information is very important. Hence, with the use of Atrous-convolution, network is allowed to enlarge the field of view of filters to incorporate larger context. Though it is simple yet powerful technique to extract the accurate localization (small-view of field) with contextual information of larger view. With the use of Atrous-convolution larger-output of feature map without increasing number of parameters can be

achieved. For semantic segmentation larger-feature map with wider field of view is required for better results. In standard convolution, pooling is performed to have smaller output feature map. Continuous pooling and increasing striding may lead to lose the spatial information in deeper layer. However in Atrous convolution our stride rate is Constant and field of view is larger without increasing its parameters. So, by avoiding this, in deeper layers we have larger feature map that is very helpful for semantic Segmentation.

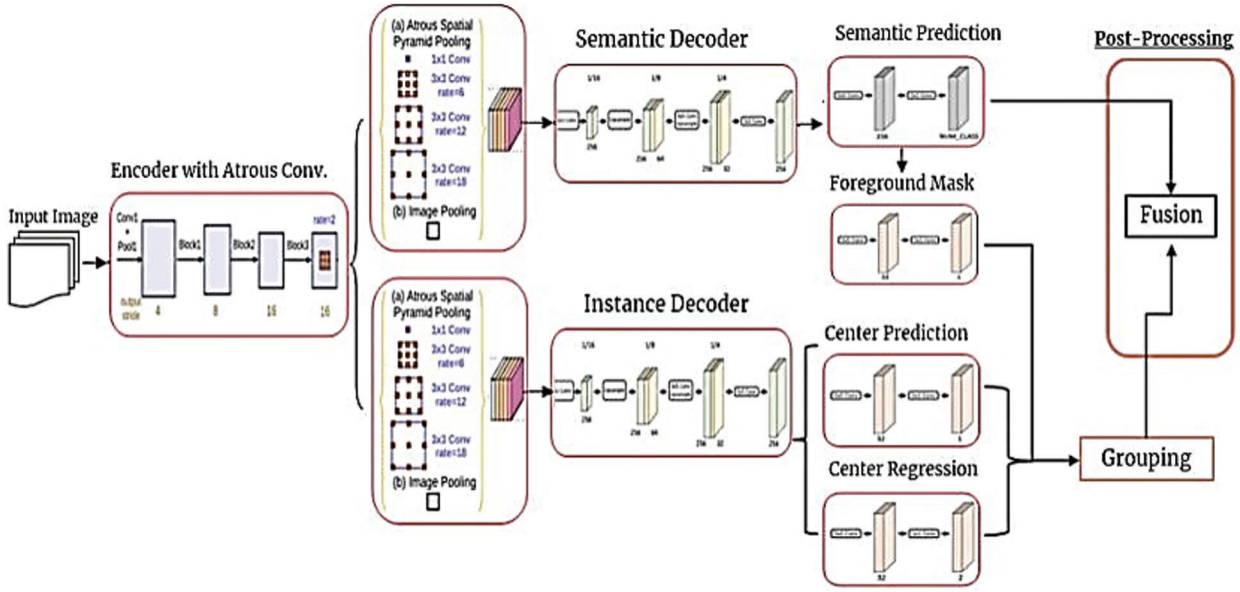

Figure 5: Proposed Architecture

MobileNetV2 is based on inverted residual structure which uses light weight depth wise convolutions in its intermediate layers to filter features. Hence, it is better due to the combination of Atrous Convolution as it does work in depth-wise convolution to incorporate larger contextual information.

5.2 ASPP (Atrous Spatial Pyramid Pooling)

Atrous Spatial Pyramid pooling modules are used for both the tasks of semantic and instance segmentation. ASPP is actually advanced version of Spatial Pyramid Pooling (SPP) module which adopts Atrous convolution with different rates at input feature map. In image different objects of same class occurs at different scales, so ASPP helps to obtain multi-scale incorporate this. In this architecture, multiple Atrous convolutions are working parallel and then combined together. In ASPP all the max-pooling operations are replaced with depth wise separable convolution to increase its computational efficiency. Two individual modules for both the semantic and instance segmentation task are employed, because these two segmentation approaches require different contextual-information that is further processed in their decoders.

5.3 Decoder (DeepLabV3+)

The decoder architecture of [27] with some convolutional modification is used, Deeplabv3+ architecture. According to the dataset. 1) 5x5 depth wise separable convolution on every up-sampling layer is applied. 2) Single convolutional is applied on every up-sampling layer with an output stride of 8, due to this, spatial resolution is gradually recovered by factor of 2. In decoder part spatial information is gradually restored that is lost in encoding due to reduction in spatial dimension. So, in Deeplabv3+ same

size of feature map is used in decoder-part that is used prior in encoding, for more accurate boundary segmentation like U-Net architecture of encoder-decoder.

In this architecture depth-wise separable convolutional (DWSC) approach is used instead of normal convolution operation. In depth-wise convolution, each filter channel is considered as a one separate input channel. For example, if we have 3-channel filter with3-channel image, the new layer will break our filter and image into 3-separate channels and perform convolution on the corresponding filters and channels, after that they are stacked back. But depth wise separable convolution is little bit different in sense that it considers depth (width) and spatial dimension (height) of a filter separately. After separable convolution, point-wise convolution (separable part of DWSC) of 1x1 filter is used. With the use of depth-wise separable convolution the output channels are same but many of parameters are reduced. Another advantage of using this convolution is that it saves us from over-fitting.

5.4 Semantic Head

In the semantic head weighted bootstrap cross entropy loss function is used to train on pixel-wise class probability at threshold of 0.7. Semantic prediction is aimed to minimize the loss. In this function we sort each pixel based on the differential weights for cross-entropy loss and back propagate the error in top-K position. Top-K is the way in which highest error losses are selected. Semantic loss function is working like a hard-mining and network is able to focus on hard-pixels along with small instances. Top-K is 0.2 for highest losses threshold.

5.5 Instance Head

5.5.1 Center Prediction

Proposed instance segmentation head uses a regression approach to identify the pixels based on instance. All the pixels of instance are associated from its instance-centroid. For this purpose, we use L1-loss function to identify the parts of objects to define the instance. Each individual pixel uses a score to learn instance vector for each pixel coordinate. Pixels coordinates points the centroid of the instance. Then each object instance is represented by its centroid. To calculate the loss of the offset 0.01 is set as an offset threshold.

$$\sum_{i=1}^{n} |y_{true} - y_{predicted}| \tag{1}$$

5.5.2 Center Regression

Heat-map is used for center regression. It predicts the pixels within a radius of the key points corresponding to the centroid's pixels for accurate heat-map. In proposed method, radius is kept same for the predicted key-points regardless of size of instance. Due to this, network learn both the small and large instances. Ground truth instance-centroids are encoded by 2D-Gausian using standard deviation, during training. Finally, we use the Mean Square Error (MSE) loss to penalize the predicted Heat-map and encoded ground truth Heat-map. MSE is ideal for regression-based tasks. This loss is sum of the square between predicted and actual value. 0 value of MSE means model is perfect or nearest to zero is considered to be a better one. We try different values ranges from (160–220) and set 180 as a center threshold. Further details are mentioned in experimentation section.

$$\text{MSE} = \frac{\sum_{i=1}^{n} (y_i - y_i^p)^2}{n} \tag{2}$$

5.5.3 Foreground Predication

After prediction of centroid, offset position of every foreground pixel (things) is predicted corresponding to its mass center. For the filtration of semantic pixels, semantic segmentation prediction is used, whose instance id is always 0. Cross entropy loss function is used to all the pixels that are belongs to the object instance based on the predicted offset of its corresponding mass center.

Most of the instance segmentation architectures use clustering approaches to combine the predicted pixels, but proposed method has not used any clustering mechanism. In proposed method all the foreground pixels are grouped together based on their nearest predicted instance-centroid, to achieve a class-agnostic instance segmentation approach. Details about the threshold and evaluation of instances are further defined in experimentation section. For filtering the foreground pixels from predicted semantic segmentation region (stuff) same threshold of 0.7 like semantic loss function has been used but having Top- k = 0.1 threshold to more focus on small instances along with large.

5.6 Post-Processing (Panoptic)

In proposed post-processing method for panoptic segmentation, Majority voting mechanism is used. Both the semantic and instance prediction have Stuff and things pixels along with ids. From semantic segmentation stuff classes pixels are used and a unique instance label is assigned. From instance segmentation we use things class pixels. Corresponding pixels of things in semantic and stuff in instance predictions are also resolved using majority voting mechanism. This parallelizable approach is effective and efficiently implemented using GPU's.

6 Experiments

6.1 Dataset

Cityscape's dataset consisting of diverse urban street scenes across 50 different German cities at different times of the year has been used to evaluate the proposed model. The dataset includes several months and seasons (spring, summer, and fall). Cityscapes provide dense pixel level annotations (Fine annotation) of 5000 frames at (1024 $\times$ 2048) resolutions. Every 20th image has been annotated from every 30 frame video snippets. In Oct, 2019 this dataset was updated to support the new panoptic challenge. Proposed method used the fine annotation of this dataset that consist of 5000 frames consisting of 30-classes including 2975 training, 500-validation and 1525-testing images. After prepossessing the structure of dataset changes it maps data into 19-classes including 8-thing classes (instance-based) and 11-stuff classes (semantic-based). Test set of datasets is not available publicly. For the purpose of benchmarking, they evaluate approaches using their own evaluation servers. We used training and validation set for all the experiments and testing purpose.

6.2 Evaluation Metrics

6.2.1 Semantic Segmentation

In order to evaluate the semantic prediction, Intersection Over-Union (IOU) has been used. IOU computed the overall pixel that belongs to certain class in an image. They treat all the pixels in an image as a single mask and ignore the object instances.

$$\frac{1}{class}\sum_{c}\left\{ IOU = \frac{Areo\ of\ overlap}{Area\ of\ Union} = > \frac{TP_c}{[TP_c - FP_c - FP_c]} \right. \tag{3}$$

Similarly for better evaluation of the segmentation and to overcome the overshooting problem we use Frequency Weighted Intersection over Union (FW-IOU). By weighting per class IOU based on the frequency of the class region.

$$\sum_{c} \frac{P_c}{P} \quad \frac{TP_c}{TP_c + FP_c + FN_c} \tag{4}$$

6.2.2 Instance Segmentation

Here, we borrow the standard evaluation metric for instance segmentation. We use Average Precision (AP) as an evaluation metrics. Basically, AP is the area under the precision- recall curve. AP is calculated over multiple intersection over union (IOU) thresholds.

$$\text{mAP} \ = \frac{1}{classes} \sum_{classes} \frac{TP_c}{TP_c + FP_c} \tag{5}$$

6.2.3 Panoptic Segmentation

Panoptic segmentation has its own evaluation metric that is introduced in 2019. Panoptic Quality (PQ) is the evaluation metric used for the challenges of panoptic segmentation. Basically, panoptic quality is a product of Segmentation Quality (SQ) and Recognition Quality (RQ). In SQ we capture the average of match segments. And in RQ we collect the information about correctly detected objects.

$$\text{PQ} = \underbrace{\frac{\sum_{(p,g)\epsilon TP} IOU_{(p,g)}}{|TP|}}_{\text{Segmentation quality}} \times \underbrace{\frac{|TP|}{|TP| + \frac{1}{2}\,|FP| + \frac{1}{2}\,|FN|}}_{\text{Recognition quality}} \tag{6}$$

6.3 Implementation Details

In proposed model we perform training using transfer-learning technique, the backbone network is MobileNet-V2 model that is pertained on the ImageNet database. This model is publicly available on PyTorch platform. DeeplabV3+ with encoder-decoder architecture is used as an architecture in the proposed model. There are four different losses that are used to optimize during the training stage. In semantic head weighted bootstrap cross entropy with threshold of 0.7 is used and bootstrap weight of back-propagation rate for highest losses threshold is Top K= 0.2. For instance segmentation, center loss using Mean Square Error loss function having 180 as a center threshold is used. L1-loss function performs offset loss at a rate of 0.01 as a threshold, then foreground loss function use cross-entropy loss with threshold of 0.7. In proposed model Adam Optimizer along with Multistep-LR learning policy at 0.001 initial rate is utilized. The code was implemented on Google Colab using AWS GPU's resources of two NVidia Tesla M60 GPU available with 16GB memory with a total of 3 images per batch size.

Adopted dataset contains 2975-training images at original resolution of (1024 × 2048) pixel size. No cropping is applied. There are total 19-classes involves, 11-stuff and 8-things. We use official split ratio of train/Val set that includes 2975/500 images respectively.

7 Results and Discussion

In this section of results, we describe our achieved scores of different experiments with different iterations. Performance on different parameters of training will be described in further sections. We achieved our best scores at 95 K iteration of training using 3-batch size. The results are presented in

Tab. 1. We evaluate our method with panoptic quality (PQ), recognition quality (RQ), and segmentation quality (SQ).

In Tab. 1 Cityscapes validation set results has been enumerated. Without using extra data (i.e., only Cityscapes fine an notation), Panoptic-DeepLab (MobileNet-V2) achieves 59.27% PQ, which is the far better than the other panoptic architectures [21,23,25]. As compared to others we utilize the minimum resources and our instance segmentation and semantic segmentation results are respectively better than other Panoptic approaches. Note that we do not exploit any other data, such as cityscapes coarse annotations, depth, or video. Panoptic segmentation includes semantic and instance but due to very limited amount of training dataset the instance segmentation start learning after 30–40 K iterations. So, we train our architecture at most 120 K iteration and find the best results shown in Tab. 1. Foreground weights are very important for instances learning. Cityscape's dataset has imbalance classes issue if we use higher weights for some classes (Train, Bus, Car) that has more mass index achieves better scores but others are ignored and predicted badly, shown in Tab. 3, we find 180-rate of foreground that perform better for all the classes.

Table 1: Results of the baseline model on Cityscapes validation for different values of training iterations based on MobileNet-V2 with original crop size of 1025x2049

Dataset	Batch-Per image	Iterations	Semantic segmentation		Instance segmentation		Panoptic segmentation		
			MIOU	FIOU	mAP	mAP-50	PQ	SQ	RQ
Validation Set	Single Batch Using 1-GPU Data Loader- 2 Foreground 200	90 K	64.79%	77.59%	17.30%	31.76%	53.81%	67.93%	60.77%
	Two Batch Using 1-GPU Data Loader- 2 Foreground 200	90 K	69.91%	81.39%	19.79%	37.82%	55.47%	71.01%	65.47%
	Three Batch Using 2-GPU's Data Loader- 4 Foreground 180	90 K	73.86%	88.92%	21.78%	38.84%	58.01%	76.77%	68.51%
		95 K	**77.27%**	**91.18%**	**23.17%**	**44.21%**	**59.27%**	**79.89%**	**71.34%**
		100 K	79.04%	92.65%	23.21%	42.03%	59.27%	81.92%	71.29%

Tab. 2 shows the comparison with the state of the art architectures that perform panoptic segmentation using cityscapes dataset. In [21–23,25] different crop sizes at different training iteration are used to achieve a result but the proposed method outperforms in terms of PQ. In proposed method better performance has been achieved while utilizing less computation resources. Panoptic segmentation is a High-computational task but we implement MobileNet-V2 that has fewer trainable parameters as compared to others. Tab. 2 shows that proposed method achieves far better than others.

In Fig. 6, achieved results has been shown for qualitative evaluation. if instance, semantic and panoptic segmentation. It can be observed that the detection and classification performance is very encouraging. Semantic segments boundaries are more precise. In case of panoptic segmentation, the instance masks are better aligned to objects and more precisely compares the instance masks from the instance head, due to adding more context and background information. The visual results are briefly described in Tab. 1. In Tab. 2, we compare our performance with the state of the art models.

Table 2: Comparison with the state-of-the-art methods on Cityscapes validation-set split

Model	Dataset	Training iterations	Resources utilized	Panoptic segmentation (PQ)	Semantic quality (SQ)	Regression quality (RQ)
[21]- 2020 Retina-Net (Resnet-50) FPN	Cityscapes (V) Pre-Training (ImageNet)	200 K	4-Batch Size Crop Size- 512 × 1024 24-GB single GPU	46.7%	—	—
			Original size - 1024 × 2048	55.1%	48.3%	63.6%
[22]- 2019 Mask RCNN ResNet-101 (RPN) FPN	Cityscapes (V)	12 K	16-Batch Size Original Size- 1024 × 2048 16-GPU's (distributed)	59.3%	79.7%	73%
			ResNet-101 (COCO-pretrained)	61.8%	81.3%	74.8%
[23]- 2019 Mask RCNN ResNeXt-101 (FPN)	Cityscapes (V) Pre-Training (Image-Net)	65 K	32-Batch Size Crop Size- 512 × 1024 8-GPU's (4 img/gpu)	58.1%	52.0%	62.5%
Our Proposed MobileNet-V2 DeepLabV3+	**Cityscapes (V) Pre-Training (Image-Net)**	**95 K**	**3-Batch Size Original Size- 1024 × 2048 2-GPU's (Total 16-GB size) Foreground Weight- 180 Data Loader- 4**	**59.27%**	**79.89%**	**71.34%**
[25]- 2019 Mask RCNN ResNet-50 (RPN) Pyramid Pooling Module	Cityscapes (V) Pre-Training (Image-Net)	—	2-Batch Size Crop size- 512 × 1024 12-GB Single GPU	45.9%	74.8%	58.4%

8 Ablation Experiments

We perform ablation studies on Cityscapes val. with our model based on MobileNet and Deeplab. We study the effectiveness of our modules by adding them one-by-one to the baseline. In the experiment, two different types of optimizers are used to train the model. Manually tuning of the model is impossible in deep-learning due to huge number of parameters. So, optimizer will tune the parameters according to its losses.

Adam and SGD optimizers has been employed to train the model using momentum. Momentum is used to accelerate the optimizer process and 0.9 is set as a momentum rate that work very well using same learning rate and policy without weight decay. Basically, Adam keeps past gradient along with current to optimize the loss. And momentum will help to keep the average of past gradient instead of just past gradient value. It calculates the adaptive learning rate of each parameter individually and due to this its performs better than SGD. SGD performs frequent updates, resultantly chances of high variance in model parameters based on the iterations, slowly reduces the loss function. The best value of SGD learning rate is 0.01. Semantic and Instance scores using optimizers are shown in Tab. 3.

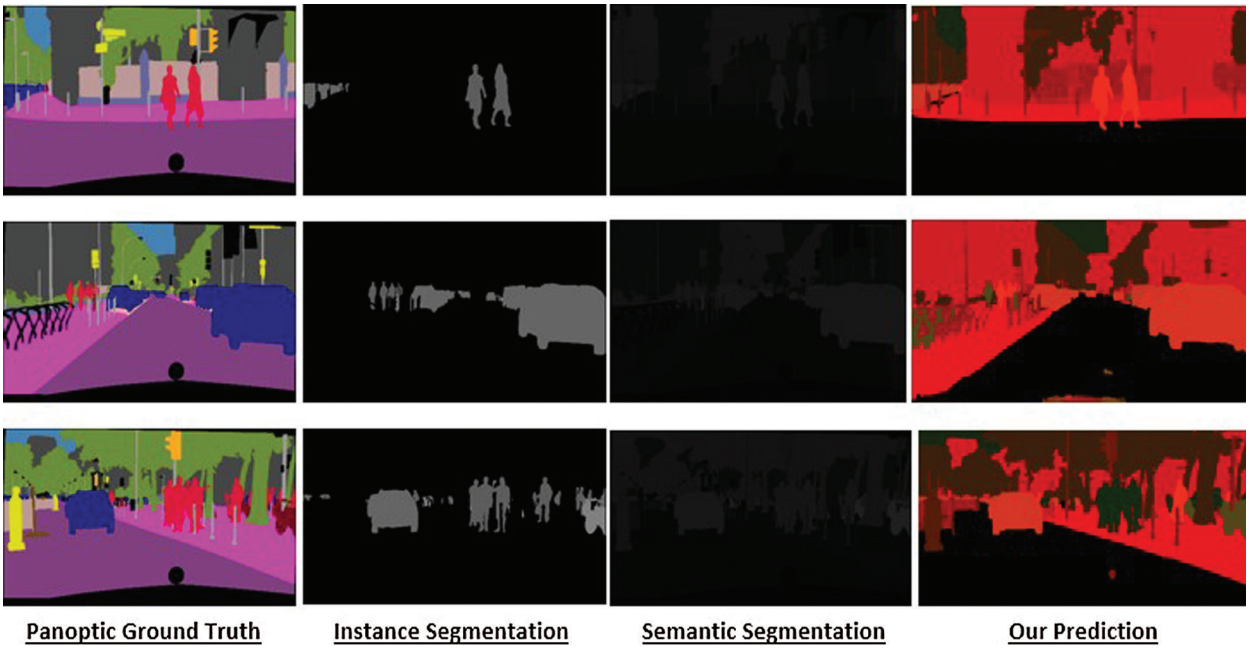

Figure 6: Visual results of Panoptic segmentation. Each instance is colored with a different shade of the same color

Table 3: Analysis of Adam and SGD optimizer scores

Data Used	Optimizer	Iterations	MIOU	mAP
Validation set	Adam	70 K	62.48	0.127
	SGD	70 K	48.29	0.052

Foreground weights are very important for instances. Weights are changed at different rates. Using less weights performs well on small instance classes but lose the large instances and model work on small regions of large instances. So, our hit and trail method find 180 weight which is an average weight that performs better than others. Further details are mentioned in Tab. 4.

Learning rate Schedule is used to adjust the learning rate (loss) of model during training according to the predefined schedule. There are many types of schedules but three most successful schedules of classification and segmentation are used to perform testing. First one is Multistep-LR Scheduler, in this scheduler after some milestones each parameter is updated using multiplicative factor of decay known as Gamma. Milestones in this scheduler is epochs or no. of iteration. This scheduler works e best in our task,. Gamma was set to 0.1 as its default value. Second scheduler is Cusine-LR Scheduler and its learning policy is based on regular cycles. This scheduler needs minimum and maximum value of learning rate along with its period time. A comparative analysis of scheduler is shown in Tab. 5.

Table 4: Comparing of foreground weights for instance segmentation scores

Data used	Thing classes	Instance segmentation				
		Subset-1 mAP Weight = 160	Subset-2 mAP Weight = 170	Subset-3 mAP Weight = 180	Subset-4 mAP Weight = 200	Subset-5 mAP Weight = 220
	Person	0.133	0.139	**0.239**	0.237	0.209
Validation	**Rider**	0.109	0.102	**0.081**	0.073	0.067
Set	**Car**	0.351	0.369	**0.427**	0.439	0.434
	Truck	0.299	0.308	**0.353**	0.382	0.386
	Bus	0.361	0.377	**0.436**	0.444	0.448
70 K	**Train**	0.211	0.214	**0.209**	0.201	0.198
Iterations	**Motorcycle**	0.099	0.055	**0.054**	0.034	0.002
	Bicycle	0.94	0.065	**0.055**	0.031	0.008

Table 5: Analysis of learning rate scheduler

Data used	Scheduler	Optimizer	Iter.	MIOU	mAP
Valid set	**Multistep-LR**	Adam	**70 K**	**62.48**	**0.127**
	Poly-LR		70 K	60.31	0.109
	Cusine-LR		70 K	28.89	0.017

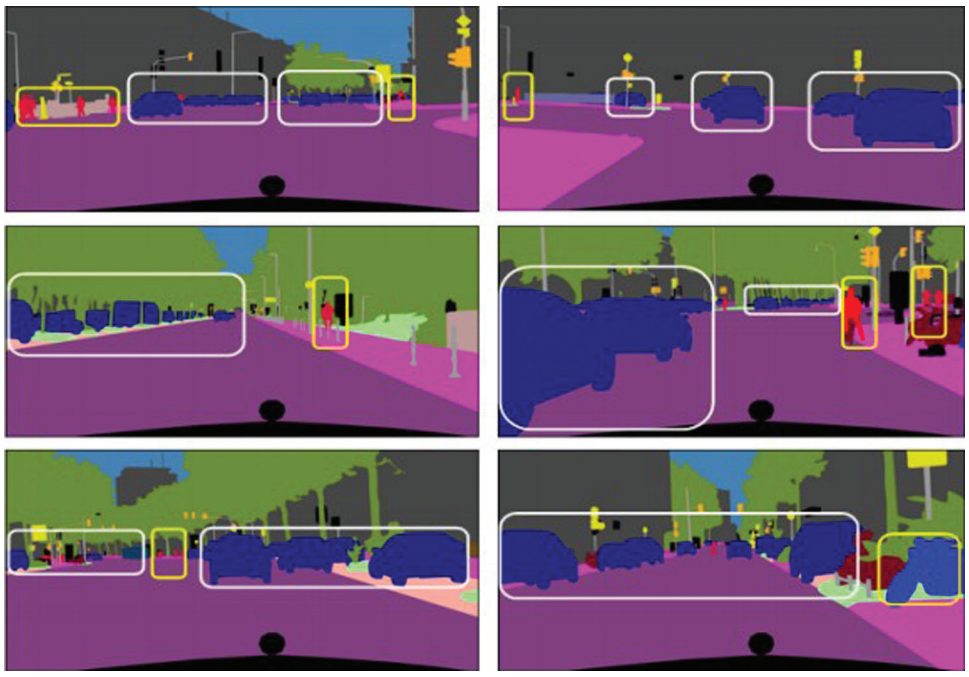

Figure 7: Cityscapes imbalance dataset

9 Conclusion

Scene understanding is very important task and has many application areas like robotics, autonomous vehicles, medical etc.

Separate approaches for semantic and instance segmentation not fulfill the needs of scene understanding and face the problems of fuzzy boundaries, over-segmentation of regions, poor localization of regions etc. In panoptic segmentation these issues can be overcome by combining both the approaches of semantic and instance. As, in panoptic segmentation, every pixel is assigned one semantic label and one instance id. The proposed architecture used MobileNet-v2 model as a backbone network with state-of-art DeepLab+ decoder. Three different loss functions for subtasks of semantic and instance instead of using different models for subtasks. Weighted cross entropy is used for semantic segmentation and perform class-agnostic instance segmentation using means square error and L1-loss. Finally, these tasks re combined using majority voting mechanism of their confidence scores and evaluated on PQ-metrics. The Backbone Network is pre-trained on ImageNet and perform training and testing on Cityscape's dataset. The proposed architecture achieves a scores of 59.27% PQ on validation- set at 95 K iterations. As compared to others existing architectures we adopt a simple architecture that work on different loss function for different sub-tasks and then jointly collaborate their prediction to achieve the final outcome. By using Panoptic Segmentation loss functions the class imbalance problem of dataset along with other problems like fuzzy boundaries has been addressed reasonably. Our backbone network (MobileNet-V2) is also a light weight network that is designed for mobile devices or embedded devices, so proposed architecture is very effective and achieves a competitive result while requires lower resources.

Cityscape's dataset is collected from real-world driving scenes of street, due to this, different classes have different ratio of appearance in the frames. So, classes are not balanced as described in Fig. 7. For further research cityscapes dataset need more attention and this limitation needs to be overcome.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] K. Alexander, R. Girshick, K. He and P. Dollár, "Panoptic feature pyramid networks," in *IEEE Conf. on Computer Vision and Pattern Recognition*, California, CA, USA, pp. 6399–6408, 2019.

[2] P. Andreas, K. Schulz and K. Dietmayer, "Semantic segmentation of video sequences with convolutional LSTM," in *IEEE Symp. on Intelligent Vehicles*, Paris, France, pp. 1441–1447, 2019.

[3] C. L. Chieh, G. Papandreou, I. Kokkinos, K. Murphy and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs, " *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834–848, 2017.

[4] A. Dundar, K. Sapra, G. Liu, A. Tao and B. Catanzaro, "Panoptic-based image synthesis," in *IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, Seattle, WA, USA, virtual event, pp. 8070–8079, 2020.

[5] S. Qiao, Y. Zhu, H. Adam, A. Yuille and L. C. Chen, "Vip-deeplab: Learning visual perception with depth-aware video panoptic segmentation," in *IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, virtual event, pp. 3997–4008, 2021.

[6] J. Lazarow, K. Lee, K. Shi and Z. Tu, "Learning instance occlusion for panoptic segmentation," in *IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, Seattle, WA, USA, Virtual event, pp. 10720–10729, 2020.

[7] R. Hou, J. Li, A. Bhargava, A. Raventos, V. Guizilini *et al.,* "Real-time panoptic segmentation from dense detections," in *IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, Seattle, WA, USA, virtual event, pp. 8523–8532, 2020.

[8] Z. Yi, K. Sapra, F. A. Reda, K. J. Shih, S. Newsam *et al.,* "Improving semantic segmentation via video propagation and label relaxation," in *IEEE Conf. on Computer Vision and Pattern Recognition*, California, CA, USA, pp. 8856–8865, 2019.

[9] K. Maninis, C. Sergi, C. Yuhua, J. P. Tuset, L. L. Taixé *et al.,* "Video object segmentation without temporal information," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 6, pp. 1515–1530, 2018.

[10] L. Jiangyun, Y. Zhao, J. Fu, J. Wu and J. Liu, "Attention-guided network for semantic video segmentation," *IEEE Access*, vol. 7, pp. 140680–140689, 2019.

[11] P. Andreas, K. Schulz and K. Dietmayer, "Semantic segmentation of video sequences with convolutional LSTM," in *IEEE Symp. on Intelligent Vehicles*, Paris, France, pp. 1441–1447, 2019.

[12] J. Samvit, X. Wang and J. E. Gonzalez, "Accel: A corrective fusion network for efficient semantic segmentation on video," in *IEEE Conf. on Computer Vision and Pattern Recognition*, California, CA, USA, pp. 8866–8875, 2019.

[13] Z. Yi, K. Sapra, F. A. Reda, K. J. Shih, S. Newsam *et al.,* "Improving semantic segmentation via video propagation and label relaxation," in *IEEE Conf. on Computer Vision and Pattern Recognition*, California, CA, USA, pp. 8856–8865, 2019.

[14] W. Tinghuai. "Context propagation from proposals for semantic video object segmentation," in *IEEE Int. Conf. on Image Processing (ICIP)*, Megaron, Athens, pp. 256–260, 2018.

[15] S. Mennatullah, S. Valipour, M. Jagersand and N. Ray, "Convolutional gated recurrent networks for video segmentation," in *IEEE Int. Conf. on Image Processing (ICIP)*, Beijing, China, pp. 3090–3094, 2017.

[16] C. Yadang, C. Hao, A. Liu and E. Wu, "Multilevel model for video object segmentation based on supervision optimization," *IEEE Transactions on Multimedia*, vol. 21, no. 8, pp. 1934–1945, 2019.

[17] C. L. Chieh, G. Papandreou, I. Kokkinos, K. Murphy and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834–848, 2017.

[18] L. Xiaoyu, H. Ren and T. Ye, "Spatio-temporal attention network for video instance segmentation," in *IEEE/CVF Int. Conf. on Computer Vision Workshops*, Seoul, Korea, pp. 1–10, 2019.

[19] F. Qianyu, Z. Yang, P. Li, Y. Wei and Y. Yang, "Dual embedding learning for video instance segmentation," in *IEEE/CVF Int. Conf. on Computer Vision Workshops*, Seoul, Korea, pp. 20–25, 2019.

[20] Y. Linjie, Y. Fan and N. Xu, "Video instance segmentation.," in *IEEE Int. Conf. on Computer Vision*, Seoul, Korea, pp. 5188–5197, 2019.

[21] D. Geus, D. P. Meletis and G. Dubbelman, "Fast panoptic segmentation network," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1742–1749, 2020.

[22] X. Yuwen, R. Liao, H.Zhao, R. Hu, M. Bai *et al.,* "Upsnet: A unified panoptic segmentation network," in *IEEE Conf. on Computer Vision and Pattern Recognition*, California, CA, USA, pp. 8818–8826, 2019.

[23] K. Alexander, R. Girshick, K. He and P. Dollár, "Panoptic feature pyramid networks," in *IEEE Conf. on Computer Vision and Pattern Recognition*, California, CA, USA, pp. 6399–6408, 2019.

[24] L. Huanyu, C. Peng, C. Yu, J. Wang, X. Liu *et al.,* "An end-to-end network for panoptic segmentation," in *IEEE Conf. on Computer Vision and Pattern Recognition*, California, CA, USA, pp. 6172–6181, 2019.

[25] D. G. Daan, P. Meletis and G. Dubbelman, "Single network panoptic segmentation for street scene understanding," in *IEEE Intelligent Vehicles Symp.*, Paris, France, pp. 709–715, 2019.

[26] M. Rohit and A. Valada, "Efficientps: Efficient panoptic segmentation," *International Journal of Computer Vision*, vol. 129, no. 5, pp. 1551–1579, 2021.

[27] C. Bowen, M. D. Collins, Y. Zhu, T. Liu, T. S. Huang *et al.,* "Panopticdeeplab: A simple, strong, and fast baseline for bottom-up panoptic segmentation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Seattle, WA, USA, pp. 12475–12485, 2020.