MBS-NET: A MOVING-CAMERA BACKGROUND SUBTRACTION NETWORK FOR AUTONOMOUS DRIVING

Jianli Wei[1], Jinwei Jiang[2], Alper Yilmaz[1]

[1]Photogrammetric Computer Vision Lab., The Ohio State University, Columbus, OH, USA - (wei.909, alper.15)@osu.edu
[2]Ford Motor Company - jjiang30@ford.com

Commission II, WG 7

KEY WORDS: Background Subtraction, Moving Camera, Conditional Random Fields, Convolutional Neural Networks.

ABSTRACT:

Background subtraction aims at detecting salient background which in return provides regions of moving objects referred to as the foreground. Background subtraction inherently uses the temporal relations by including time dimension in its formulation. Alternative techniques to background subtraction require stationary cameras for learning the background. Stationary cameras provide semi-constant background images that make learning salient background easier. Still cameras, however, are not applicable to moving camera scenarios, such as vehicle embedded camera for autonomous driving. For moving cameras, due to the complexity of modelling changing background, recent approaches focus on directly detecting the foreground objects in each frame independently. This treatment, however, requires learning all possible objects that can appear in the field of view. In this paper, we achieve background subtraction for moving cameras using specialized deep learning approach, the Moving-camera Background Subtraction Network (MBS-Net). Our approach is robust to detect changing background in various scenarios and does not require training on foreground objects. The developed approach uses temporal cues from past frames by applying Conditional Random Fields as a part of the developed neural network. Our proposed method have a good performance on ApolloScape dataset (Huang et al., 2018) with resolution 3384×2710 videos. To the best of our acknowledge, this paper is the first to propose background subtraction for moving cameras using deep learning.

1. INTRODUCTION

Primary goal of background subtraction is to find moving objects based on their differences from the salient background which is learned from a stream of images. This task can be considered as classification of each pixel as background or foreground that can be designated as a pixel-wise binary semantic segmentation task. Generalizing this binary segmentation to more than two classes, aka. semantic background subtraction, has shown to improve the performance of background subtraction. These methods aim to finally label pixels into a number of moving object regions (Cioppa et al., 2020, Braham et al., 2017). Apart from background subtraction, another body of work directly detect the objects for each frame (Yokoyama and Poggio, 2005). These object detection methods are widely used in low-level computer vision tasks such as video surveillance, robotics and authentication systems. Modern object detection methods seek to locate object instances by learning predefined categories from images (Liu et al., 2020, Redmon et al., 2016, Redmon and Farhadi, 2017, Redmon and Farhadi, 2018). Considering that there can be thousands of different object categories, these methods cannot be generalized. Increasing these known object categories indefinitely not only increases computational cost but also the complexity of the model used to learn these categories. In fact, for some problems, knowing the category of the object is not be important: for autonomous vehicle scenarios, obstacles on the roads are important and they can be in any form from among thousands of different categories of objects one can think of. Hence, distinguishing between background and foreground becomes more important and is a liability for autonomous driving. Especially, during autonomous driving, the vehicle cameras provide images that contain background region build up of sky, buildings, lanes, trees and the road itself among others. One can arguably conjecture that the number of categories appearing in the background is significantly limited compared to that of object categories. Therefore, learning the background, hence the background subtraction is a more efficient and feasible task. In this work, we model the background using convolutional neural networks (CNN) that has been successfully applied to image segmentation among others to model complex and recessive relationships between the inputs and outputs (Vemulapalli et al., 2016).

Labeling pixels as background or foreground may introduce spurious regions and noisy labels that can be reduced by imposing spatial and temporal regularization. Conditional Random Fields (CRFs) has been used generally for spatial regularization purposes as a probabilistic graph model (Lafferty et al., 2001). While there have been other regularization solutions introduced in the past, such as Hidden Markov Models (HMMs) (Krogh et al., 2001) and stochastic grammars (Zhu et al., 2007), CRFs offers a directed probability graph model to relax strong and causal dependence assumptions between current frame and its previous adjacent frame(s). In this study, we introduce the CRFs as temporal regularization to ensure the background information learned in the previous frames is carried over to the current CNN output. This introduction is critical as the past data contains important temporal cues that help refine the current result which in return would improve the background subtraction accuracy. Additionally, adopting CRFs as a CNN layer would keep the end-to-end solution of the CNN based approaches.

Another important consideration in background subtraction is the loss function minimized. In a typical Gaussian Mixture Model, maximum likelihood minimization estimates the mix-

ing parameters and the model parameters. In neural networks a number of generic differentiable loss functions have been introduced. While they generally work well, they do treat hard to learn or easy to learn examples the same. *Focal Loss* introduced in (Lin et al., 2017) is an advanced loss function by reshaping the standard cross entropy loss. *Focal Loss* assigns dynamic weights to all samples. For easy examples with high confidence, it down-weights the loss. This treatment avoids easy to learn examples that are dominant and makes the loss function focus on training hard examples so that to accelerate network convergence and improve network accuracy at the end. Hence, in our approach we adopt Focal Loss for learning process

Our main contributions to background subtraction are summarized as follows:

- Most background subtraction approaches require stationary cameras. Our approach is among the few that performs background subtraction for moving-camera where the salient image regions constantly change;
- We introduce a temporal regularization through CRFs layer to rectify the end-to-end CNN solution by modeling interactions between previous frames and current CNNs output. CRFs layer in MSN-Net has improved the performance from 68.35% to 76.06% on foreground and 96.64% to 97.53% on background.
- We apply *Focal Loss* to assign all training samples dynamic weights to avoid easy examples establishing dominance over the loss.
- We propose MBS-Net that achieves impressive results on the benchmarks of ApolloScape dataset. More specifically, we achieve 97.53% Mean IoU on background and 76.06% on foreground.

The rest of this paper is organized as follows. Section 2 reviews recent related work in background subtraction. Section 3 states the problem we have on current approaches. Section 4 detailed illustrate our proposed MBS-Net. Section 5 shows our experiment results and the ablation study of our proposed MBS-Net.

2. RELATED WORK

Background subtraction has been an active area of research for a long time. There are several baseline methods such as the Gaussian Mixture Model (Zivkovic, 2004), Principle Component Analysis (Guyon et al., 2012) and its variants, Kernel Density Estimation (Mittal and Paragios, 2004), and Mean Shift (Piccardi, 2004). While these techniques have been used over the past two decades they do not typically apply spatial and/or temporal regularization to labeled pixels. Zamalieva (Zamalieva et al., 2014) introduced the motion, appearance, temporal and spatial regularization in the labeling cost which they minimized using graph-cut. While their method works with nominal camera motion it suffers from larger camera motions due to the optical flow estimation step that requires small camera motion.

Aside from more traditional background subtraction algorithms, deep learning have also been used in more recent papers. These techniques build on the development of CNNs over the past decade. While there is a large body of work on object detection and tracking in the published literature, we will cite only a few as representatives of their categories as they relate to background subtraction. Considering background subtraction provides regions of moving objects, one can use deep learning to detect and track the objects directly. Wang (Wang et al., 2019) proposed object tracking aiming at predicting trajectories of multiple targets in video sequences. Girdhar (Girdhar et al., 2018) performs object detection in video by building on achievements in human detection and video understanding. Alternative to object detection and tracking another approach one can consider is semantic scene segmentation. In (Long et al., 2015, Ronneberger et al., 2015, Chen et al., 2017), authors semantically segment an image which provides enclosing boundaries of the objects in the scene as well as the clutter region which can be vaguely considered as the background. While the goal is different, video scene parsing can also be considered as a way of detecting objects in the video. Among others, scene parsing can be performed based on optical flow estimation (Gadde et al., 2017, Kroeger et al., 2016), recurrent neural networks (Hochreiter and Schmidhuber, 1997, Fayyaz et al., 2016) and convolutional networks (Shelhamer et al., 2016).

For most training datasets, imbalance between easy and hard examples, as well as, negative and positive examples are common problems. These two problems always happen for datasets that can be used in moving-camera background subtraction where the background mostly is composed of sky and buildings. A common treatment is to modify and customize loss function to focus the loss on the harder training examples. In recent years, some novel approaches have been proposed to release this imbalance problem including Online Hard Example Mining (Shrivastava et al., 2016), gradient harmonizing mechanism (Li et al., 2019), and Focal Loss (Lin et al., 2017) introduced by Lin *et al.*. Those proposed loss functions are originally proposed for object detection, which has serious imbalance as only few proposals contain objects over millions of candidate proposals. Therefore, we here firstly introduce their approaches to our background detection problem. Experiments indicate it benefits our training process as well as the network predicted output.

Our proposed MBS-Net in the paper, modifies the BiSeNet network architecture(Yu et al., 2018). The BiSeNet generates one-stage output from two paths within the architecture, the spatial path and the context path, to preserve image size and provide large receptive field. MBS-Net shares three main modifications. First, we use *Focal Loss* during training training process, which distributes dynamic weights for all examples and makes hard examples to be dominant during training. Then we introduce upsampling of fused feature map(s) back to the original input size. Finally, we add CRFs to the network to achieve temporal regularization by sharing previous frames labeling constraint with current CNN output.

3. PROBLEM STATEMENT

We seek to achieve background subtraction by using vehicle embedded cameras. As the vehicle moves, the camera sees a new scene and the background changes. The motion of the camera is dependent on the vehicle motion which can be forward or backward while turning or going straight. These motion types provides geometric conditions on the types of images acquired. In Figure 1, we show an example road scene acquired from a vehicle mounted camera. The same figure also shows the background reference in gray color where the labeling criteria for background include sky, buildings, road, lane signs and trees. The remaining regions including pedestrians, vehicles in the

reference image correspond to other objects indicated as foreground.

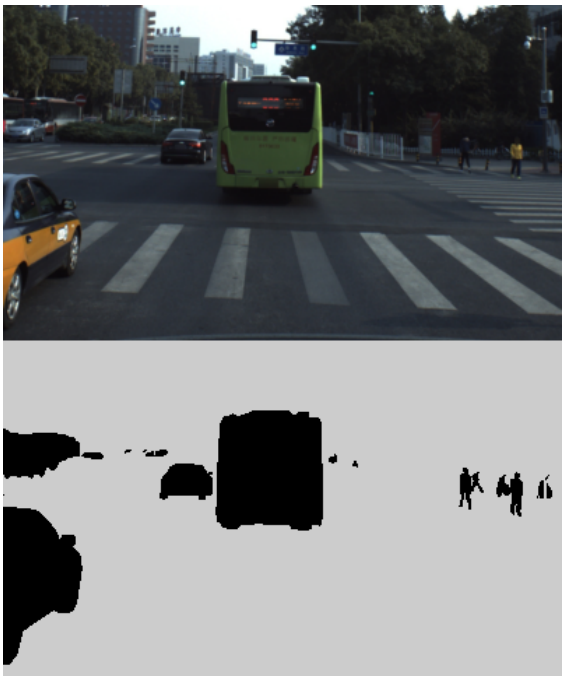


Figure 1. Top image is an example frame acquired from a camera mounted on a vehicle showing the road scene from vehicle's perspective. The bottom image is a background reference with pixel-wise labels; gray part indicates background and black part indicates foreground.

A sequence of frames acquired from the vehicle camera contains redundancy and temporal cues that provide important constraints on the solution. Fusing the temporal cues mainly introduces two advantages. First, temporal cues carry semantic information, when utilized would improve the background subtraction performance. Second, it regularizes the labeling process and provides coherency that smooths the generated labels in time axis. The proposed semantic segmentation network, MBS-Net, and others in the literature, do not consider regularization in spatial and time domains. The vanilla application of these approaches to the background subtraction problem, where the acquired image sequence contains visual shakes, creates many spurious regions that are incorrectly identified as background or foreground. These spurious labels also are observed due to other reasons which can be removed using the temporal information. Based on these advantages of the temporal information, MBS-Net introduces the temporal regularization in the background estimation step.

4. MOVING BACKGROUND SUBTRACTION NETWORK ARCHITECTURE

The proposed MBS-Net has Convolutional Neural Network BiSeNet (Yu et al., 2018) as its backbone architecture. The use of CRF introduces the temporal regularization to background estimation and overcomes spurious regions and the effects of camera shakes during vehicle is in motion. MBS-Net also adopts *Focal Loss* in order to tackle the sample imbalance problem.

4.1 Convolutional Neural Network Architecture

The MBS-Net is built on the BiSeNet architecture. The BiSeNet includes a *spatial path*, a *context path* , an *attention refinement module* and a *feature fusion module*. In this paper, the temporal regularization is introduced by modifying this architecture. The structure of the MBS-Net architecture is illustrated in Fig. 2.

In this architecture, the spatial path preserves the spatial size of the original input image. It extracts feature maps that are 1/8 of the original image size by cascading three 2D convolution layers with $stride = 2$. Therefore, the spatial path encodes spatial information with many details preserved in large sizes feature maps. In contrast to spatial path, the context path perceives sufficient details for large receptive fields at the pixel level. With respect to the consideration of having large receptive fields versus high computational cost, we adopted light-weight models of Xception(Chollet, 2017) and MobileNet(Howard et al., 2017). Once the embedding is completed, the attention refinement module refines the extracted context by integration of spatial and contextual information. This step is followed by the feature fusion module that integrates the two paths and encodes the integrated feature maps back to input image size.

4.2 Conditional Random Fields

The architecture introduced in Fig. 2 contains a specialized fully connected layer operating on the time axis to ensure consistency in labeling for the image sequence. This fully connected layer acts as CRF that models the interaction between the current frame and a set of previous frames. In context of deep learning, CRFs have been used to improve spatial model interactions between respective input image and output labels (Chen et al., 2017). While there are similarities, our approach to CRF is different and they model the interactions in time axis. This application requires changes to the model and the kernels used. Specifically, the kernels in our work become:

$$k(f_i, f_j) = w^{(1)} exp(-\frac{|P_i - P_j|^2}{2{\Theta_\alpha}^2} - \frac{|l_i - l_j|^2}{2{\Theta_\beta}^2}) + w^{(2)} exp(-\frac{|P_i - P_j|^2}{2{\Theta_\gamma}^2}) \quad (1)$$

where the first kernel models temporal interactions where the pixels (denoted as P) and past labels (denoted as l) are used; and the second kernel models spatial interactions. The hyper parameters Θ_α, Θ_β, Θ_γ control the "scale" of the kernels and remains constant on training. One can observe that the larger these parameters are, the more likely the corresponding features get ignored. $w^{(1)}$ and $w^{(2)}$ are compatibilities, deciding how much is learnt between the two separate kernels. The larger the compatibility is, the more its corresponding kernel gets weighted.

4.3 Focal Loss

Modern object detection mainly has two sub-branches including one-stage and two-stage approach. In two-stage approach, the first stage generates a sparse set of candidate proposals and the second stage classifies them. One-stage approach combine the above two stages as one at the same time. Thus one-stage object detection has an extreme candidate location imbalance between object and non-object samples on training. These detectors would evaluate $10^4 - 10^5$ candidate-proposals but only few of them contain objects. Similar to the dilemma one-stage

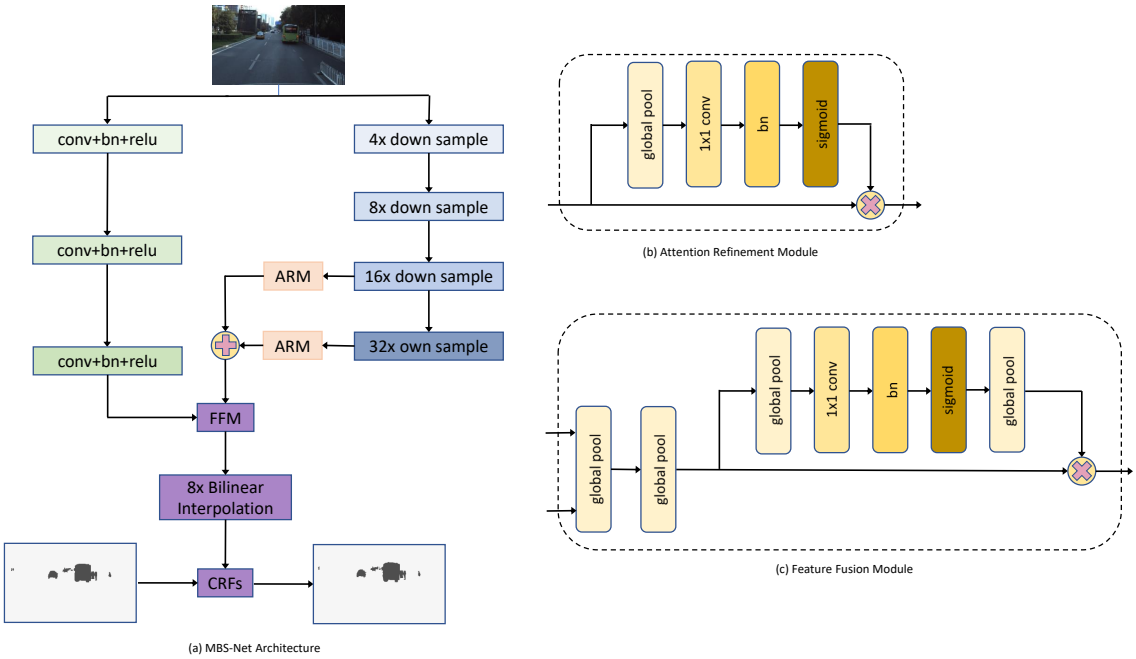


Figure 2. Overview of MBS-Net Architecture. (a) MBS-Net Structure including temporal regulation; (b) attention refinement module structure (c) feature fusion module structure.

object detection faced with, we also need to tackle the sample imbalance problem. In our training dataset, the ratio of background and foreground samples reach $10:1$ or so. The *focal Loss* is originally designed for one-stage object detection (Lin et al., 2017), which has a better trade-off between speed and accuracy compared with two-stage methods. Traditionally, weighted cross-entropy loss function in (2) is applied in classification problem:

$$CE(p_k) = -\alpha_k log(p_k) \tag{2}$$

where p_k ($p_k \in$[0,1]) represents softmax probability of the sample belonging to ground-truth class k and α_k ($\alpha_k \in$[0,1]) specifies predefined weight of class k. While α balances the contribution of background and foreground examples, all samples in the same class are still of same significance, no matter how easy/hard those samples are. As training going further and deeper, the easily classified pixels grow up to be the majority of the loss and will dominate the back-propagation gradients. This could impede even stop neural networks learning dataset. Lin *et al.* (Lin et al., 2017) proposed *Focal Loss* which introduced dynamic weights for all samples to reshape the loss as:

$$FL(p_k) = -\alpha_k(1-p_k)^\gamma log(p_k) \tag{3}$$

where γ is focus parameter $\gamma \geq 0$. *Focal Loss* is able to down-weight easy pixels as probability p_k get close to 1. For example, when γ=2, a pixel k classified with p_k=0.9 would contribute the loss just 1/100 compared with that in CE loss. The pixel contribution keeps almost same when $p_k \to 0$. Therefore, by reshaping CE loss, *Focal Loss* could adjust hard-classified pixels to be dominant of the loss. Specially, FL loss is equivalent to CE loss when γ=0.

5. EXPERIMENTS

In our implementation, we introduced the Xception39 into Spatial Path of BiSeNet. Using this code with other changes including *Focal Loss*, *CRFs* temporal regulation layer and *Deconvolution* upsampling technique, we evaluate its performance on ApolloScape road02_seg dataset, which is avaiable from ApolloScape website. It contains 25 snippets and 11435 continuous frames in total. We manually divided them into training, validation and testing datasets, which respectively have 7923, 1200 and 2312 frames. Besides, all frames are fine annotated have a resolution of $3,384 \times 2,710$, in which each pixel is annotated to 25 different predefined labels by 8 groups, listed in Table 1. In Section 5.1, we introduce *ApolloScape* dataset and provide experiment details. In section 5.2, we report Mean-Intersection-over-Union (mIoU) accuracy and frame(s)

Category	Class
Sky	sky
Movable objects	car, car group
	motorbicycle, motorbicycle group
	bicycle, bicycle group
	person, person group
	rider, rider group
	truck, truck group
	bus, bus group
	tricycle, tricycle group
Flat	road
	sidewalk
Road obstacles	traffic cone
	road pile
	fence
Roadside objects	traffic light
Void	pole
	traffic sign
	wall
	dustbin
	billboard
Building	building
	bridge
	tunnel
	overpass
Natural	vegetation

Table 1. Class Definitions

per second (fps) speed results on the ApolloScape testing dataset. At the end, Section 5.3 investigates the effect of fully connected CRFs, Focal Loss and Bilinear Interpolation Upsampling approaches by ablation study.

5.1 Implementation details

ApolloScape datasets are first released by *Baidu Research* containing 140K time dependent images and corresponding semantic pixel-level labels. These datasets are collected in various cities in recent years in China, aiming to increase its variability and complexity of urban street views(Huang et al., 2018). Each frame is acquired one meter apart with the equipped vehicle keeping velocity of 30 km/h. All frames in each snippet are time dependent. Considering that the goal of this paper is to detect background and foreground regions from images acquired by a moving camera, we fuse all classes into background and foreground; such as sky, building and road become background and denoted as 1; and everything else including all moving objects become foreground and are labeled as 0.

In our tests, traditional mean subtraction and standard normalization methods are not used due to the fact that the batch normalization(Ioffe and Szegedy, 2015) layers normalize feature maps inside the mini-batches. There are over $11k$ frames in our preprocessed ApolloScape dataset. This rich set of fine-annotated frames removes the typical requirement for traditional data augmentation such as random flip and random crop. Therefore, we only employ a sequential crop and resize operations including cropping frame resolution from 3384×2710 to 960×1600 and resizing it to 240×400 to keep the object shapes, cut the computational cost and save GPU memory.

Using this dataset, we implemented the MBS-Net architecture which contains three convolutional layers with *stride*=2 in its Spatial Path, and a pretrained Xception39 model in Context Path. The model uses Attention Refinement Module and Feature Fusion Module (FFM) to refine and fuse feature maps generated from the two paths. The output of FFM is 1/8th of the input image. The bilinear interpolation is then used to enlarge the output map back to the original image size. Finally , CRF layers are enforced as a temporal regularization layer within MBS-Net and they model interactions between current frame *t* and previous *n* frames. In the experiment, with the consideration of short-time dependency and long-time independency of video frames, we set *n*=1 and repeat the boundary frames within each snippet. Note that, the CRFs layer is only activated during testing.

Our implementation uses Adam optimizer with initial *learning_rate* $\eta_0 = 3e^{-3}$, and applies "step-wise decay" learning rate strategy into training process, where the initial learning rate decay with power 0.9 every 2 epochs $\eta = \eta_0 * 0.9^{\lfloor \frac{n}{2} \rfloor}$. The *Focal Loss* is initialized with α_0=0.75, α_1=0.25 and γ=1. Mini batch size is set *batch_size*=8 due to GPU memory limitation.

In testing phase, we use *use_crf* to decide whether activating CRFs layer or not. Once *use_crf*=True, CRFs would be employed with predefined hyperparameters $w^{(1)}$=0.5, $w^{(2)}$=3.5, Θ_α=Θ_γ=(2,2) and Θ_β=2. Those hyperparameters are obtained by our empirical studies.

We note that training and testing implementations are conducted with PyTorch on NVIDIA Titan V.

5.2 Results

The computational bandwidth autonomous vehicles is constraint due to other tasks the vehicles is performing every second. Hence, the speed becomes a key factor to algorithm evaluation. Aside from the quantitative comparisons, for this stated reason, we conduct experiment to compare different backbones architectures shown in Table 2. In our test, the fastest results are optained at $305 fps$ using ResNet18 as the base-model.

In Table 2, we compare the speed of MBS-Net with several different popular basemodels. We count MBS-Net total parameters under each basemodel, as well as its speed with and without activating CRFs layer respectively. All experiments are conducted on NVIDIA Titan V.

Base-Model	Params Num	Speed[1] (fps)	Speed with CRFs[2] (fps)
GoogleNet	8,784,290	162.3	3.2
Xception39	28,019,986	187.2	3.3
MobileNet	4,279,034	190.7	3.2
ResNet18	12,410,754	305.3	3.6
ResNet34	22,518,914	224.5	3.2
ResNet50	31,246,466	184.0	3.3
ResNet101	50,238,594	114.6	3.3

[1] These results are testing the speed without activating fully connected CRFs.
[2] Fully connected CRFs layer is activated and final output is evaluated on current network output and previous 1 frame.

Table 2. Speed Analysis

For all experiments in Table 2, input image has the resolution of 240*400 for fair comparison. In this speed experiment, we don't apply any loss function or measure matrix for simulating real-scene practice.

Aside from achieving high throughput, we have also achieved the state-of-the-art accuracy in quantitative analysis. Among the variants of ResNet basemodels, we pick ResNet50 as it outperformed others in the experiments. We have also tested GoogleNet, MobileNet and Xception39 as a part of the MBS-Net architecture, and selected Xception39 as in our final design. In order to have a fair comparison, we test ResNet50 and Xception39 basemodel MBS-Net on the above mentioned test dataset and compute Mean IoU with activating and deactivating CRFs layer, shown in Table 3.

In Table 3, we assess the accuracy within two best performed basemodels, ResNet50 and Xception39. Mean IoU of background and foreground is computed with activating and deactivating CRFs layer under the two MBS-Net basemodels. It can be observed that the CRFs significantly improves foreground detection for the Xception39 architecture (highlight row). We may also notice that CRFs layer slightly improves foreground detection for ResNet50 architecture. This is because ResNet50 based MBS-Net is more powerful on detecting boundaries between foreground and background, and CRFs as temporal regulation approach contributes mainly on boundaries regulation in the same way.

5.3 Ablation study

This section demonstrates the performance of several components including *CRFs* layer, *Focal Loss* and *Bilinear Interporation Upsampling*. To deliver a fair and clear comparison, Xception39 is used as the basemodel of MBS-Net and ApolloScape road02_seg test dataset is the evaluation dataset.

BaseModel	Mean IoU(%)[1]		Mean IoU(%) w/. CRFs	
	BG[2]	FG[3]	BG	FG
Xception39	97.38	68.35	97.46	76.06
ResNet50	97.75	75.08	97.71	75.78

[1] Without CRFs
[2] Background
[3] Foreground

Table 3. Accuracy Analysis

CRFs is a necessary part of the MBS-Net and has been originally used for image semantic segmentation without considering temporal regularization. There is a large improvement, CRFs in MBS-Net has improved the performance from 68.35% to 76.06% on foreground and 96.64% to 97.53% on background, as shown in Table 3 highlight row. Quantitatively, these results are important as shown in Fig.3 and Fig.4.

To overcome dataset imbalances, and improving accuracy, MSB-Net assigns dynamic weights for all examples. For some hard examples with low probability leading to misclassification, we assign higher weights than other easy examples. Compared with traditional crossentropy loss, *Focal Loss* shares two main advantages based on our experiments. On the one hand, it accelerates our MBS-Net convergence reflecting on test Mean IoU. With *CE Loss* and *Focal Loss*, MBS-Net respectively achieves 73.33% and 74.56% on foreground after first 10 training epochs. On the other hand, *Focal Loss* achieves higher Mean IoU by efficiently mining hard examples. Compared to *CE Loss*, *Focal Loss* improves the performance from 75.07% to 76.06%.

Our final ablation study is on Bilinear Interpolation Upsampling. Upsampling layer is designed to increase resolution of fused feature maps to the original input image. Some existing approaches include *Bilinear Interpolation, pooling indices memorization, deconvolution, etc*. Different from the other two approaches, *pooling indices memorization* requires sharing those pooling indices from encoder feature map(s) with corresponding feature map(s) in decoder (Badrinarayanan et al., 2017). As discussed in Section 4.1 stated, Spatial Path (SP) in MBS-Net cascades three *Conv+BN+Relu* blocks with *stride*=2 so which downsamples images to 1/8th of input image. Hence, *pooling indices memorization* will not be an alternative approach because of the loss of pooling indices information.

Table 4. Bilinear Interpolation Upsampling Ablation Analysis

Upsampling Approach Approach	Additional Params[1]	Mean IoU[2] (%)	Speed[3] (fps)
Bilinear Interpolation	-	68.35	187.2
Deconvolution	66	66.80	165.5

[1] Additional params represents increased parameters num compared to *Bilinear Interpolation* approach.
[2] Accuracy performance is evaluated on foreground category only.
[3] Speed is evaluated without activating CRFs layer.

Here we mainly compare the performance of *Bilinear Interpolation and deconvolution*, as shown in Table 4. Bilinear Interpolation outperforms Deconvolution approach both on speed and accuracy without introducing additional parameters.

CONCLUSIONS AND FUTURE WORK

In this paper, we introduced MBS-Net that modifies an existing semantic segmentation CNN architecutre by including additional steps and layers to ensure temporal regularization is performed in the background labeling process. The temporal regularization step combined with spatial regularization have been tested on the ApolloScape benchmark dataset and is shown to achieve good results. We apply *Focal Loss* which reshapes cross entropy loss in order to focus on hard to learn examples during training. We also design ablation study to investigate their efficacy showing that it can achieve state-of-the-art accuracy and speed. Reward mechanisms such as the ones used in reinforcement learning is an ongoing research.

ACKNOWLEDGEMENTS

This project was funded by the Ford Motor Company.

REFERENCES

Badrinarayanan, V., Kendall, A., Cipolla, R., 2017. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12), 2481–2495.

Braham, M., Piérard, S., Van Droogenbroeck, M., 2017. Semantic background subtraction. *2017 IEEE International Conference on Image Processing (ICIP)*, 4552–4556.

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A. L., 2017. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4), 834–848.

Chollet, F., 2017. Xception: Deep learning with depthwise separable convolutions. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1251–1258.

Cioppa, A., Droogenbroeck, M. V., Braham, M., 2020. Real-Time Semantic Background Subtraction. *ArXiv*, abs/2002.04993.

Fayyaz, M., Saffar, M. H., Sabokrou, M., Fathy, M., Huang, F., Klette, R., 2016. Stfcn: spatio-temporal fully convolutional neural network for semantic segmentation of street scenes. *Asian Conference on Computer Vision*, Springer, 493–509.

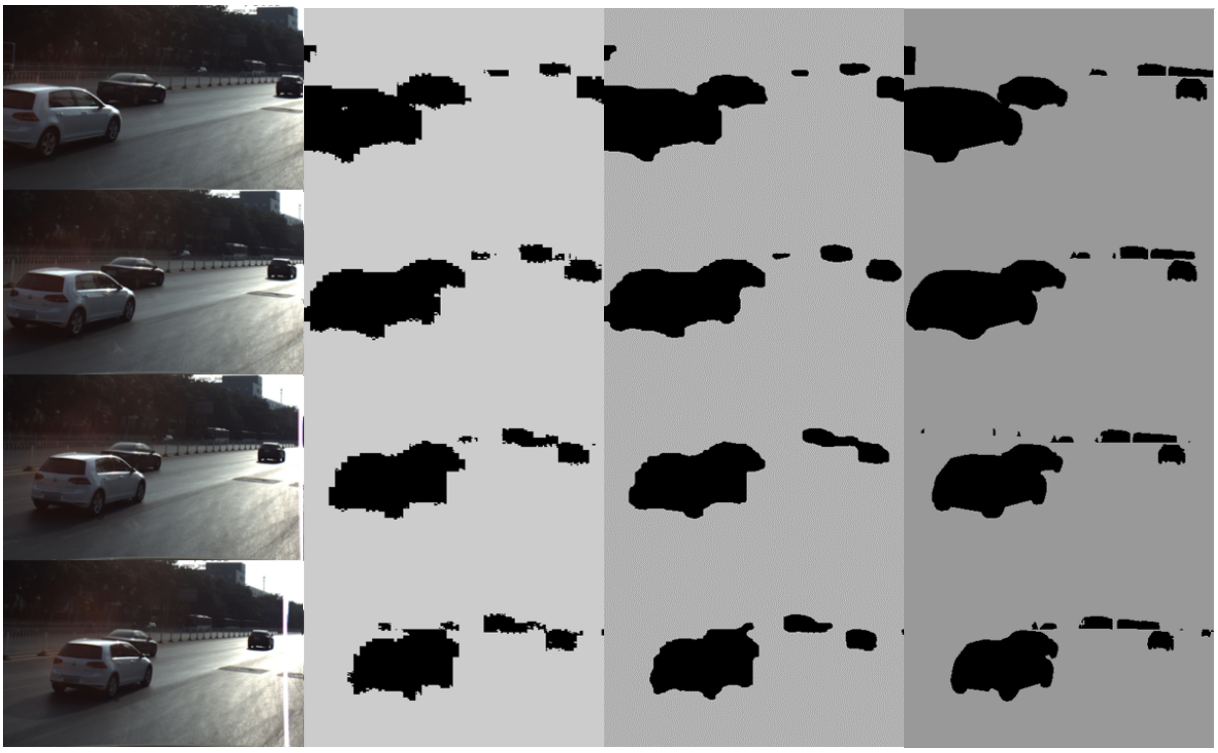
Gadde, R., Jampani, V., Gehler, P. V., 2017. Semantic video cnns through representation warping. *Proceedings of the IEEE International Conference on Computer Vision*, 4453–4462.

Girdhar, R., Gkioxari, G., Torresani, L., Paluri, M., Tran, D., 2018. Detect-and-track: Efficient pose estimation in videos. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 350–359.

Guyon, C., Bouwmans, T., Zahzah, E.-h., Sanguansat, P., 2012. Robust principal component analysis for background subtraction: Systematic evaluation and comparative analysis. *Principal component analysis*, 10, 223–238.

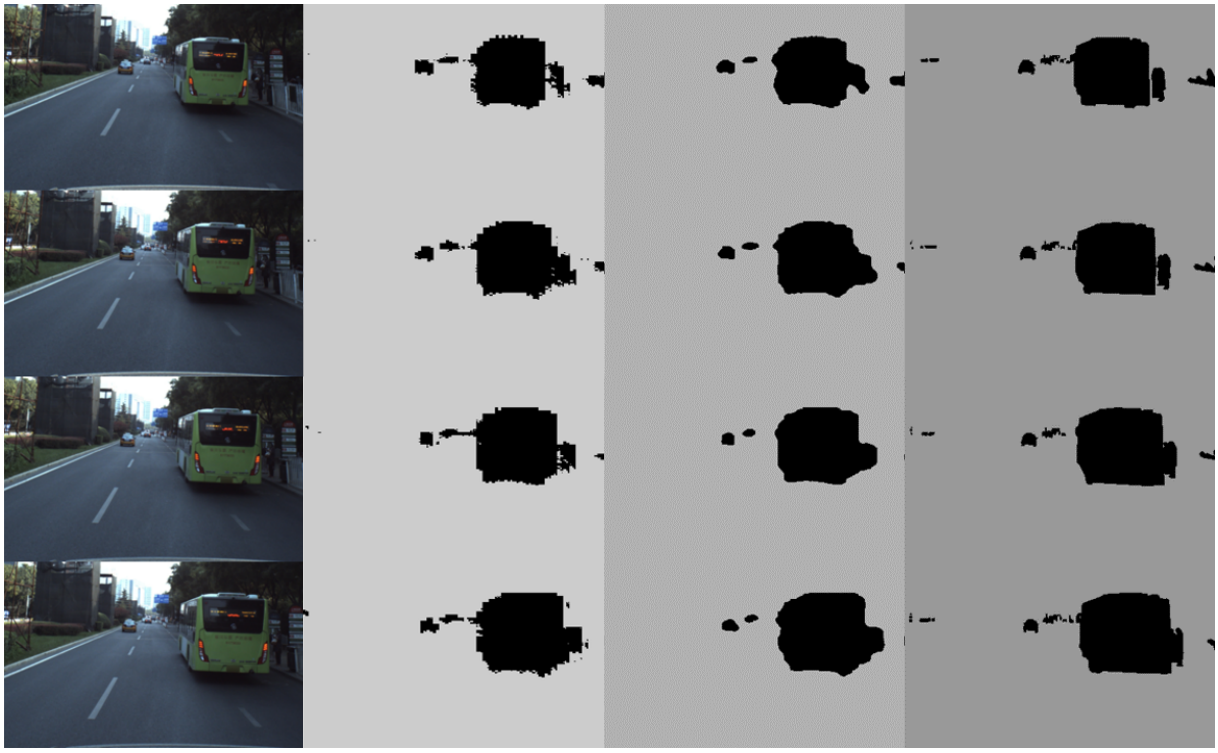
Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. *Neural computation*, 9(8), 1735–1780.

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H., 2017. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.



(a) Image (b) MBS-Net (c) MBS-Net with CRFs (d) GT

Figure 3. MBS-Net example results (a) is original input sequential images. (b) is sequential MBS-Net output masks without activating CRFs (c) is sequential output masks with CRFs (d) is Ground Truth



(a) Image (b) MBS-Net (c) MBS-Net with CRFs (d) GT

Figure 4. MBS-Net example results (a) is original input sequential images. (b) is sequential MBS-Net output masks without activating CRFs (c) is sequential output masks with CRFs (d) is Ground Truth

Huang, X., Cheng, X., Geng, Q., Cao, B., Zhou, D., Wang, P., Lin, Y., Yang, R., 2018. The ApolloScape Dataset for Autonomous Driving. *arXiv: 1803.06184*.

Ioffe, S., Szegedy, C., 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*.

Kroeger, T., Timofte, R., Dai, D., Van Gool, L., 2016. Fast optical flow using dense inverse search. *European Conference on Computer Vision*, Springer, 471–488.

Krogh, A., Larsson, B., Von Heijne, G., Sonnhammer, E. L., 2001. Predicting transmembrane protein topology with a hidden Markov model: application to complete genomes. *Journal of molecular biology*, 305(3), 567–580.

Lafferty, J., McCallum, A., Pereira, F. C., 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

Li, B., Liu, Y., Wang, X., 2019. Gradient harmonized single-stage detector. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33, 8577–8584.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., Dollár, P., 2017. Focal loss for dense object detection. *Proceedings of the IEEE international conference on computer vision*, 2980–2988.

Liu, L., Ouyang, W., Wang, X., Fieguth, P., Chen, J., Liu, X., Pietikäinen, M., 2020. Deep learning for generic object detection: A survey. *International journal of computer vision*, 128(2), 261–318.

Long, J., Shelhamer, E., Darrell, T., 2015. Fully convolutional networks for semantic segmentation. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3431–3440.

Mittal, A., Paragios, N., 2004. Motion-based background subtraction using adaptive kernel density estimation. *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, 2, Ieee, II–II.

Piccardi, M., 2004. Background subtraction techniques: a review. *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No.04CH37583)*, 4, 3099–3104 vol.4.

Redmon, J., Divvala, S., Girshick, R., Farhadi, A., 2016. You only look once: Unified, real-time object detection. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 779–788.

Redmon, J., Farhadi, A., 2017. Yolo9000: better, faster, stronger. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7263–7271.

Redmon, J., Farhadi, A., 2018. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation. *International Conference on Medical image computing and computer-assisted intervention*, Springer, 234–241.

Shelhamer, E., Rakelly, K., Hoffman, J., Darrell, T., 2016. Clockwork convnets for video semantic segmentation. *European Conference on Computer Vision*, Springer, 852–868.

Shrivastava, A., Gupta, A., Girshick, R., 2016. Training region-based object detectors with online hard example mining. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 761–769.

Vemulapalli, R., Tuzel, O., Liu, M.-Y., Chellapa, R., 2016. Gaussian conditional random field network for semantic segmentation. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3224–3233.

Wang, Z., Zheng, L., Liu, Y., Wang, S., 2019. Towards Real-Time Multi-Object Tracking. *arXiv preprint arXiv:1909.12605*.

Yokoyama, M., Poggio, T., 2005. A contour-based moving object detection and tracking. *2005 IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, 271–276.

Yu, C., Wang, J., Peng, C., Gao, C., Yu, G., Sang, N., 2018. Bisenet: Bilateral segmentation network for real-time semantic segmentation. *Proceedings of the European conference on computer vision (ECCV)*, 325–341.

Zamalieva, D., Yilmaz, A., Davis, J. W., 2014. A multi-transformational model for background subtraction with moving cameras. *European Conference on Computer Vision*, Springer, 803–817.

Zhu, S.-C., Mumford, D. et al., 2007. A stochastic grammar of images. *Foundations and Trends® in Computer Graphics and Vision*, 2(4), 259–362.

Zivkovic, Z., 2004. Improved adaptive gaussian mixture model for background subtraction. *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, 2, IEEE, 28–31.