Improvement in object segmentation techniques over a live video sequence with optimal inference run-time

(using Deep Neural Networks)

Sambandh Bhusan Dhal
Department of Electrical and Computer Engineering
Texas A&M University
College Station, USA
sambandh@tamu.edu

Anirudh Singh Shaktawat
Department of Electrical and Computer Engineering
Texas A&M University
College Station,USA
ug201311005@tamu.edu

Arun Agarwal
Department of Electronics and Instrumentation Engineering
ITER, Siksha 'O' Anusandhan University, Khandagiri Square
Bhubaneswar-751030, Odisha, India
arunagrawal@soauniversity.ac.in

Kabita Agarwal
Department of Electronics and Telecommunication Engineering
CV Raman College of Engineering
Bhubaneswar-751054, Odisha, India
akkavita22@gmail.com

Abstract—**This paper deals with the improvement of segmentation techniques used for detecting objects in video sequences. A lot of work has been done in the past which have shown good segmentation results on still images. Here, I have tried to propose an extension of Mask R-CNN[7] for segmentation on video images with some inputs from OSVOS architecture[2] which is used to segment an object along the entire video given the mask of only one single labeled sample and the classification of all the pixels of video sequence into foreground and background given the annotation reference of one or more frames.**

Keywords-segmentation;Mask R-CNN;OSVOS;annotation

I. INTRODUCTION

The paper "Person identification using spatio-temporal characteristics" [5] has tried to address this segmentation approach as a development over other model-based video-object segmentation techniques and use different skeletal data characteristics such as joints positionings and their orientation or other grid features such as stride length, grid length and other shape templates. This algorithm computes the dense trajectories based on optical flow field and their information about motion description is encoded using local descriptors. A codebook based on Gaussian Mixture Models is built and the features are encoded using Fischer vector. The features which we encode here are classified using Support Vector Machines to recognize the individuals. This paper although has produced significant improvements in accuracy but the feature reuse and warping used in this technique using optical flow has encountered limits in the speed up of the process.

In "Fast Semantic Segmentation on Video Using Motion Vector-Based Feature Interpolation" [1], an improvement over this existing technique has been proposed. This can be done by a two-stage approach to eliminate the process of optical flow. Here, a fast feature propagation scheme is used which utilizes the block motion vector maps to cheaply propagate features from frame to frame. Secondly, they have developed a novel feature estimation scheme, termed as feature estimation, which fuses features from enclosing keyframes to render accurate estimates. In this paper, I have tried to combine the techniques used for semantic segmentation in Mask R-CNN for still images with the concept of motion vectors to detect objects in a video sequence.

II. EXPERIMENTS AND RESULTS

A. Mask R-CNN[7]:

This technique adds an additional branch for predicting an object mask with an existing branch for bounding box recognition and mask prediction which is the baseline of our work. This was performed on Cityscapes dataset which contain 2975 train, 500 validation and 1525 test images using ResNet-50 FPN[4].

There are 7 categories of images in the dataset where we reduce the learning rate at every 3000 iterations and run un-

til 4000 iterations. This gives an output of 32.0 AP on the test set which is an improvement of over six points over the fine-only counter-part. The results have been shown in Table 1.

	training data	AP [val]	AP	AP_{50}	person	rider	car	truck	bus	train	mcycle	bicycle
InstanceCut [23]	fine + coarse	15.8	13.0	27.9	10.0	8.0	23.7	14.0	19.5	15.2	9.3	4.7
DWT [4]	fine	19.8	15.6	30.0	15.1	11.7	32.9	17.1	20.4	15.0	7.9	4.9
SAIS [17]	fine	-	17.4	36.7	14.6	12.9	35.7	16.0	23.2	19.0	10.3	7.8
DIN [3]	fine + coarse	-	20.0	38.8	16.5	16.7	25.7	20.6	30.0	23.4	17.1	10.1
SGN [29]	fine + coarse	29.2	25.0	44.9	21.8	20.1	39.4	24.8	33.2	30.8	17.7	12.4
Mask R-CNN	fine	31.5	26.2	49.9	30.5	23.7	46.9	22.8	32.2	18.6	19.1	16.0
Mask R-CNN	fine + COCO	**36.4**	**32.0**	**58.1**	**34.8**	**27.0**	**49.1**	**30.1**	**40.9**	**30.9**	**24.1**	**18.7**

Figure 1: Results of Mask R-CNN on Cityscapes dataset. This method uses ResNet-50 FPN [7]

B. *One Shot Video Object Segmentation(OSVOS)[2]:*

This technique was performed on DAVIS dataset where 50 full HD Video sequences with all frames were segmented with pixel level accuracy. It was performed with some semi-supervised, unsupervised and Bounds techniques and have been shown in Table 2 and 3.

To further validate our results, a few manually annotated objects in a Youtube video were also taken into consider-ation.

		Semi-Supervised								Unsupervised							Bounds		
Measure		Ours	OFL	BVS	FCP	JMP	HVS	SEA	TSP	FST	NLC	MSG	KEY	CVOS	TRC	SAL	COB|SP	COB	MCG
$\mathcal{J}$	Mean $\mathcal{M}\uparrow$	**79.8**	68.0	60.0	58.4	57.0	54.6	50.4	31.9	**55.8**	55.1	53.3	49.8	48.2	47.3	39.3	**80.5**	79.3	70.7
	Recall $\mathcal{O}\uparrow$	**93.6**	75.6	66.9	71.5	62.6	61.4	53.1	30.0	**64.9**	55.8	61.6	59.1	54.0	49.3	30.0	**96.5**	94.4	91.7
	Decay $\mathcal{D}\downarrow$	14.9	26.4	28.9	**−2.0**	39.4	23.6	36.4	38.1	**0.0**	12.6	2.4	14.1	10.5	8.3	6.9	2.8	3.2	**1.3**
$\mathcal{F}$	Mean $\mathcal{M}\uparrow$	**80.6**	63.4	58.8	49.2	53.1	52.9	48.0	29.7	51.1	**52.3**	50.8	42.7	44.7	44.1	34.4	**87.1**	75.7	62.9
	Recall $\mathcal{O}\uparrow$	**92.6**	70.4	67.9	49.5	54.2	61.0	46.3	23.0	51.6	51.9	**60.0**	37.5	52.6	43.6	15.4	**92.4**	88.5	76.7
	Decay $\mathcal{D}\downarrow$	15.0	27.2	21.3	**−1.1**	38.4	22.7	34.5	35.7	**2.9**	11.4	5.1	10.6	11.7	12.9	4.3	2.3	3.9	**1.9**
$\mathcal{T}$	Mean $\mathcal{M}\downarrow$	37.6	21.7	34.5	29.6	15.3	35.0	**14.9**	41.2	34.3	41.4	29.1	25.2	**24.4**	37.6	64.1	**27.4**	44.1	69.8

Figure 2: DAVIS Validation: OSVOS versus the state of the art, and practical bounds[2].

Category	Ours	OFL	JFS	BVS	SCF	AFS	FST	HBT	LTV
Aeroplane	88.2	**89.9**	89.0	86.8	86.3	79.9	70.9	73.6	13.7
Bird	**85.7**	84.2	81.6	80.9	81.0	78.4	70.6	56.1	12.2
Boat	**77.5**	74.0	74.2	65.1	68.6	60.1	42.5	57.8	10.8
Car	79.6	**80.9**	70.9	68.7	69.4	64.4	65.2	33.9	23.7
Cat	**70.8**	68.3	67.7	55.9	58.9	50.4	52.1	30.5	18.6
Cow	77.8	**79.8**	79.1	69.9	68.6	65.7	44.5	41.8	16.3
Dog	**81.3**	76.6	70.3	68.5	61.8	54.2	65.3	36.8	18.0
Horse	**72.8**	72.6	67.8	58.9	54.0	50.8	53.5	44.3	11.5
Motorbike	73.5	**73.7**	61.5	60.5	60.9	58.3	44.2	48.9	10.6
Train	75.7	76.3	**78.2**	65.2	66.3	62.4	29.6	39.2	19.6
Mean	**78.3**	77.6	74.0	68.0	67.6	62.5	53.8	46.3	15.5

Figure 3: Youtube-Objects evaluation: Per-category mean intersection over union[2].

C. *Person Identification Using Spatiotemporal Motion Characteristics[5]:*

To validate this technique, the results were performed on TUM GAID database (a common gait recognition database). The codebook size k was set to 32. A total of 3370 train sequences are there in the database consisting of 305 subjects. For this approach, 10 walk sequences of each subject , total of 32 subjects, were recorded using Microsoft Kinet for both Summer and Winter seasons at 30 fps. The walk sequences consisted of Normal Walk(N), Walking with Backpack(B), Walk with Shoes(S), normal walk after time (TN), walk with back-pack after time (TB) and walk with coating shoes after time (TS). Many different techniques were used on this database but our proposed algorithm achieved an accuracy of 96.5 percent on the dataset bettering other classi-fiers. Detailed results have been shown in Table 4.

Method	N	B	S	TN	TB	TS	Avg.
GEI [17]	99.4	27.1	56.2	44.0	6.0	9.0	56.0
GEV [17]	94.2	13.9	87.7	41.0	0.0	31.0	61.4
SVIM [30]	98.4	64.2	91.6	65.6	31.3	50.0	81.4
CNN-SVM [29]	99.7	97.1	97.1	59.4	50.0	**62.5**	94.2
CNN-NN128 [29]	99.7	98.1	95.8	62.5	56.3	59.4	94.2
H2M [31]	99.4	100.0	98.1	71.9	63.4	43.8	95.5
DCS [31]	99.7	99.0	99.0	78.1	62.0	54.9	96.0
PFM [28]	99.7	99.0	99.0	**78.1**	62.0	54.9	96.0
Proposed	**99.7**	**100**	**99.7**	68.8	**71.9**	53.1	**96.5**

Figure 4:Evaluation of performance on TUM GAID database. Average score is calculated as the weighted average score of each method.[5].

D. *Fast Semantic Segmentation on Video Using Motion Vector-Based Feature Interpolation[1]:*

Here, we have considered 30 frame snippets of 50 European cities. The frame rate for this experiment was calculated to be 17fps and the image dimension was 2048X1024 pixels. The entire dataset was divided into 2975 train, 500 validation and 1525 test images. Details about the accuracy of this technique have been demonstrated in Table 5 and have been more elaborately discussed in the conclusion part as it is the state of the art technique which we wish to modify and incorporate for live video object segmentation.

		keyframe interval									
Metric	Scheme	1	2	3	4	5	6	7	8	9	10
mIoU, avg (%)	prop-flow	75.2	73.8	72.0	70.2	68.7	67.3	65.0	63.4	62.4	60.6
	prop-mv	75.2	73.1	71.3	69.4	68.2	67.3	65.0	64.0	63.2	61.7
	interp	**75.2**	**73.9**	**72.5**	**71.2**	**70.5**	**69.9**	**68.5**	**67.5**	**66.9**	**66.6**
mIoU, min (%)	prop-flow	75.2	72.4	68.9	65.6	62.4	59.1	56.3	54.4	52.5	50.5
	prop-mv	75.2	71.3	67.7	64.8	62.4	60.1	58.5	56.9	55.0	53.7
	interp	**75.2**	**72.5**	**71.5**	**68.0**	**67.2**	**66.2**	**65.4**	**64.6**	**63.5**	**62.9**
runtime (fps)	prop-flow	1.3	2.3	3.0	3.5	4.0	4.3	4.6	4.9	5.1	5.3
	prop-mv	1.3	2.5	3.4	4.3	5.0	5.6	6.2	6.7	7.1	7.6
	interp	1.3	2.4	3.4	4.2	4.9	5.4	6.0	6.4	6.9	7.2

Figure 5: Accuracy and runtimes for optical flow-based feature propagation(propflow), motion vectors-based propagation (prop-mv), and feature interpolation (interp).[1]

CONCLUSION AND FUTURE WORK

The main motivation behind my work is to reduce the inference run time while segmenting along a video sequence. For this, the idea of motion vector maps as a replacement to optical flow solves a part of this problem. In [1], this idea has been explained in detail. The entire frame is divided into 16X16 non-overlapping pixels. The best matching 16X16 pixel block is compared with the reference frame which is measured by the Mean Squared Error(MSE) between the data frames and the pair with the minimum MSE is stored in the Motion Vector Mapping as a (x,y) pair.

Then, the process of feature interpolation is done for a key frame interval of size n where a key frame interval refers to the number of frames between two key frames. For a key frame interval n, n-1 forward feature maps and n-1 back-ward feature maps are computed and all of them are fused in the end to get the optimal feature map which is then used for detection along the video sequence. This whole process when compared to that of optical flow reduces the inference run time by over 50 per cent.

The main disadvantage of this technique which I seek to solve is that in live streaming, we look for n frames ahead so here, we have to look for a small delay of 30fps while segmenting along a video sequence. This may give optimal results but while segmenting in a cityscape environment where the surroundings are rapidly changing, it is difficult for us to compute the motion vectors as the spatiotemporal characteristics of the frame changes so we have to look for alternative approaches while segmenting along a live video sequence in such an environment.

In this case, OSVOS technique[2] explained earlier can prove to be handy where we can segment along the entire video sequence given the multiple annotations of single/multiple frames in a video. In future, work needs to be done to figure out how we can combine the concept of motion vector maps with OSVOS technique so that we can segment along a live video sequence without any delay and with much less computation time

REFERENCES

[1] Samvit Jain and Joseph E. Gonzalez, Fast Semantic Seg-mentation on Video Using Motion Vector-Based Fea-ture Interpolation. 2018 IEEE Conference on Com-puter Vision and Pattern Recognition (CVPR). doi: arXiv:1803.07742v2

[2] Caelles S., Maninis K.,Pont-Tuset J.,Leal-Taixe L., Cremers D., and Gool, L. V. (2017). One-Shot Video Object Segmentation. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). doi:10.1109/cvpr.2017.565

[3] Perazzi, F., Khoreva, A., Benenson, R., Schiele, B., and Sorkine-Hornung, A. (2017). Learning Video Object Seg-mentation from Static Images. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). doi:10.1109/cvpr.2017.372

[4] Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: CVPR. (2015)

[5] Khan, M. H., Farid, M. S., and Grzegorzek, M. (2017). Person identification using spatiotemporal motion char-acteristics. 2017 IEEE International Conference on Image Processing (ICIP). doi:10.1109/icip.2017.8296264

[6] Vertens, J., Valada, A., and Burgard, W. (2017). SM-Snet: Semantic motion segmentation using deep convo-lutional neural networks. 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). doi:10.1109/iros.2017.8202211

[7] He, K., Gkioxari, G., Dollar, P., and Girshick, R. (2017). Mask R-CNN. 2017 IEEE International Conference on Computer Vision (ICCV). doi:10.1109/iccv.2017.322

[8] I. Bouchrika and M.S. Nixon, Model-based fea-ture extraction for gait analysis and recognition, in ICCV.Springer, 2007, pp. 150160.

[9] Jampani, V., Gadde, R., and Gehler, P. V. (2017). Video Propagation Networks. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). doi:10.1109/cvpr.2017.336

[10] Liu, S., Wang, C., Qian, R., Yu, H., Bao, R., and Sun, Y. (2017). Surveillance Video Parsing with Sin-gle Frame Supervision. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). doi:10.1109/cvpr.2017.114