Video object segmentation using spatio-temporal deep network

Akshaya Ramaswamy, Jayavardhana Gubbi, Balamuralidhar P.
Embedded Systems and Robotics
TCS Research and Innovation, Banglore, India
akshaya.ramaswamy@tcs.com, j.gubbi@tcs.com, balamurali.p@tcs.com

***Abstract*—Video analysis is increasingly becoming possible with improvement in hardware and deep learning algorithms. Videos contain the spatial as well as the temporal information that come closest to the real-world visual information representation. Albeit the human brain can make better decisions using spatio-temporal data, the images and video frames captured from the same standard RGB camera will vary in quality. Deep learning has resulted in extraordinary performances for image analysis. Image-based deep networks have been modified and extended to work on video, and optical flow between the frames has been utilized to capture temporal variations. There is a gap in understanding whether such networks capture the spatio-temporal information collectively. The network that can capture the information effectively should be capable of good performances despite relatively bad quality video frames. In this work, different deep network architectures are explored and their ability to capture spatio-temporal features is explored. With the understanding of the advantages and disadvantages of the network components, a new network is designed for the task of video object segmentation (VOS). The performance of the proposed network is evaluated using the DAVIS dataset for three tasks: VOS using weak supervision, zero-shot VOS and one-shot VOS. The best performance is reported in comparison to the state-of-the-art on DAVIS dataset and the robustness of the model to noisy labels is demonstrated.**

***Index Terms*—Video object detection, Video reconstruction, Self-supervised learning, spatio-temporal representation**

I. Introduction

A video is a sequence of image frames that forms a moving visual scene. This provides contextual information about the scene in the form of spatio-temporal data, which is crucial for video object detection (from low-quality frames), video event detection and behaviour analysis. The challenge lies in building a network to extract the spatio-temporal information since the temporal variation is not an image property, rather a property between image frames. Some tasks require only an overall understanding of the spatio-temporal variations, such as video action recognition. Others like video object detection require a deeper look into the frame-level features.

Convolutional neural networks (CNNs) have shown remarkable performance in many image-based tasks. Video analysis has also benefited from these image-based pre-trained models. Further, memory-based units such as recurrent neural networks (RNNs) have been widely used in time series analysis of sequential signals and are now being explored for video applications. In [1], a study is performed to understand the need for multiple frames for action recognition. It is observed that around 40% of classes in UCF101 [2] and 35% of the classes in Kinetics [3] do not require the motion information for action recognition. To demonstrate the usefulness of motion in the video, datasets such as Something-Something V2 [4] and Jester [5] are introduced. These contain videos of micro-actions performed by human subjects. Many networks such as temporal relational reasoning [6] and motion-fused frames [7], proposed on top of frame-level networks, are adapted to Something-Something V2. The drawback of such methods is that they learn the sequence of action through image-level features, but do not exactly capture the temporal features. This raises the question as to whether a network is actually capturing the spatio-temporal information, or is it trivially capturing other image-level features for a video task. The goal of our work is: a) to evaluate different architectures for spatio-temporal information capture by formulating the problem as video reconstruction; and b) to design an architecture for video object segmentation by the understanding of advantages and disadvantages of different architectural configurations. The second goal is further extended to effectively build representation for zero-shot and one-shot learning scenarios that is ultimately very useful for video-based applications, where the annotation is very challenging.

II. Related work

There are a number of works that have looked at video feature and video representation learning using interesting approaches. Video object segmentation is also quite a well-explored area, with many competitions such as DAVIS [8]. We look at the prior art in video feature learning and in video object segmentation and bring out the relevance of our work in this context.

Many inventive ways have been attempted to capture spatio-temporal features in videos. One popular method is to make use of the order of the frames, which is directly related to temporal coherence. Odd-one-out network [9] proposes an algorithm where, multiple video sequences are given as input to a multi-branched network, to find the sequence in which frames are not in order. Temporally shuffled frames are given as input to a deep network [10], which learns how to sort the sequence, while another algorithm for temporal order verification is proposed by Misra *et al.* [11]. Similar to this, instead of frame order, clip order prediction is learned by the network [12] in another work. Usage of other properties

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

has also been explored such as learning of pixel correspondences [13] and learning the optical flow between frames [14]. All these works are validated using standard datasets that are biased for image-based recognition of tasks. In the proposed approach, the auxiliary property of videos is not used, video reconstruction problem is formulated that is the ultimate test for building representation.

In applications involving frame-level analysis such as video object segmentation, frame-level convolutional networks using image pre-trained models OSVOS [15] and MaskTrack [16] have become very popular, and multiple improvements over these networks have been proposed [17] [18] [19]. These networks use the segmentation mask of the first frame to compute masks in the subsequent frames using online training. The task of unsupervised or zero-shot video object segmentation was introduced in the DAVIS challenge 2019 [20]. The similarity in the background and the temporal coherence in the foreground have been used to come up with attention-based networks for zero-shot VOS [21] [22]. A motion-based bilateral network is used to estimate the background from video segments, and this is used to segment the foreground objects [23]. In another work [24], LSTMs are used to construct a network, which can be trained end-to-end offline, for both zero-shot and one-shot VOS.

To the best of our knowledge, there is only one work on using self-supervised learning for video object segmentation [13]. This uses pixel-wise correspondence matching to train the network for feature learning, and this network is evaluated on the DAVIS dataset for one-shot VOS. We design a network, pre-trained on a large number of videos for the task of video reconstruction, and fine-tune it for video object segmentation. Our network can be trained end-to-end for both zero-shot VOS and one-shot VOS. It does not require online training using the mask of the first frame for one-shot VOS.

III. Approach

The two main objectives of this work are a) to experiment with and evaluate various deep learning components for the capture of spatio-temporal features from videos; b) to come up with an optimized self-supervised network that is fine-tuned for the task of video object segmentation. Instead of focusing on discrimination as the final goal, we approach the problem as a video reconstruction problem. The idea behind this approach is that only a network that can learn the spatio-temporal information and variations from the input frames will be able to reconstruct the frames reasonably well and this can then be extended to any pattern recognition and compression applications.

A. Problem Formulation

Consider a set of input video frames f_1, f_2,...f_N, and a video segment $V = \{F_i\}|_{i=1}^{N}$ made up of the N frames stacked together. The goal is to learn all the spatio-temporal features from the input video. We conceptualize this as a video reconstruction problem, using an encoder (**E**) - decoder (**D**) framework. We consider two ways of approaching this problem. The first approach is using a single network that captures spatial features and spatio-temporal variations. From Eq. 1, F_{ST} is this set of features learned by the spatio-temporal encoder network, and from F_{ST}, the input video V has to be recovered using the decoder.

$$F_{ST} = \mathbf{E}(V) \text{ and } V' = \mathbf{D}(F_{ST}) \tag{1}$$

The second approach is designing the encoder with two separate networks – a spatial network to extract the spatial information and a temporal network to capture spatio-temporal variations between frames. From Eq. 2, F_S, denoting spatial features, and F_T, indicating temporal features are combined to reconstruct the original frames, using the decoder network.

$$F_S = \mathbf{E}_S(V); F_T = \mathbf{E}_T(V); V' = \mathbf{D}(F_S|F_T) \tag{2}$$

Reconstruction validation is performed using $L2$ cost function and is given by:

$$cost = \|(V - V')\| \tag{3}$$

An array of deep learning component exists for achieving this objective. ResNet, I3D, LSTM and the respective variants are used as building blocks for validating our hypothesis. A number of architectural possibilities emerge due to this variety and they are elaborated in the next section.

B. Video Reconstruction Network Architecture

Motivated by multiple developments in image-based architecture, we develop four video reconstruction frameworks using convolution and LSTM layers. We make use of image pre-trained models and video pre-trained models for spatio-temporal feature extraction. For the proposed architecture, we keep the input and expected output the same with a set of ten consecutive RGB video frames of size 224×224 each. The set of 10 frames is chosen based on the available action recognition dataset where micro-action is represented by approximately 0.3 seconds of video. All our architectures are based on the auto-encoder framework; the intent is that the encoder captures spatio-temporal features, and these features are validated by video reconstruction using the decoder network.

1) Network with I3D as the base: The first network architecture (N_1) uses Inflated Inception 3D (I3D) pre-trained model [25] as shown in Figure 1. The network follows a 3D-Conv–3D-Deconv framework, and additionally uses LSTM for capturing temporal information. The input set of frames is given to the I3D model, which is pre-trained on both ImageNet and Kinetics datasets. The I3D network performs 3D convolutions on the input, thereby capturing spatio-temporal variation features of the input frames. The output is given to a bi-directional 2D convolutional LSTM ($2 \times 2 \times 2$) grid. Each block in the grid consists of two LSTM layers to capture long-term temporal information, and we tap out all the intermediate LSTM outputs. The LSTM outputs of each block are concatenated along the width, height and channel dimensions to obtain the encoder feature R. We use R to retrieve the original frames by applying 3D transpose convolution layers and 3D convolution layers.

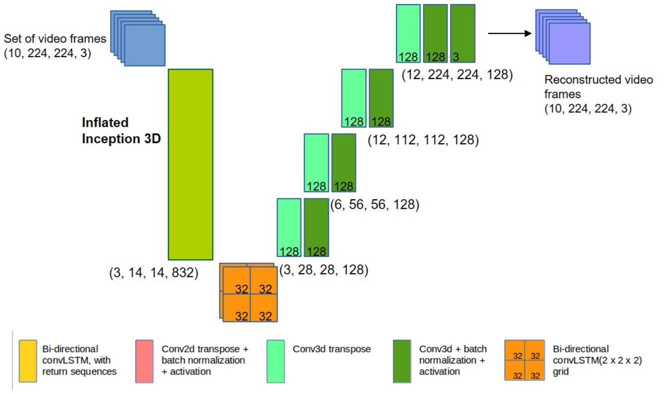

Fig. 1. Architecture (N_1) using LSTM and I3D

2) Network with ResNet as the base: The second architecture (N_2) uses ResNet as the pre-trained model and hence uses 2D convolution instead of 3D convolution as shown in Figure 2. The ResNet outputs are concatenated and input into a bi-directional 2D convolutional LSTM layer. The LSTM output R is input to 2D deconvolution layers, consisting of 2D convolution and 2D deconvolution layers to reconstruct the input. Adding deconvolution layers helps to validate how well the spatio-temporal variations are captured by the features for perfect reconstruction.

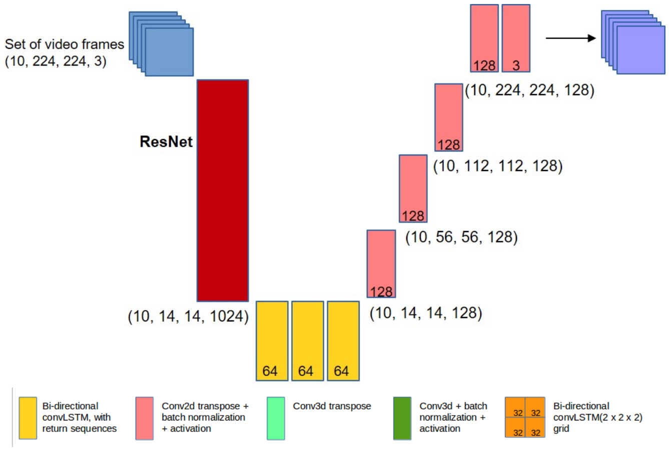

Fig. 2. Architecture (N_2) using LSTM and ResNet

3) Network with ResNet and I3D as the base: In the third variation (N_3), we make use of a combination of both I3D and ResNet. We consider the base architecture as N_1 and capture spatial and temporal properties in two ways: a) combining ResNet and I3D outputs before feeding it into LSTM (Fig. 3 top called N_{3A}); and b) combining ResNet output with LSTM output before reconstruction (Fig. 3 bottom called N_{3B}). The interpretation behind having both image and video-based pre-trained models is that spatial information is captured well by an image model, and the spatio-temporal variation is captured in the 3D architecture. Combining them can contribute to giving a more complete feature representation R of the input.

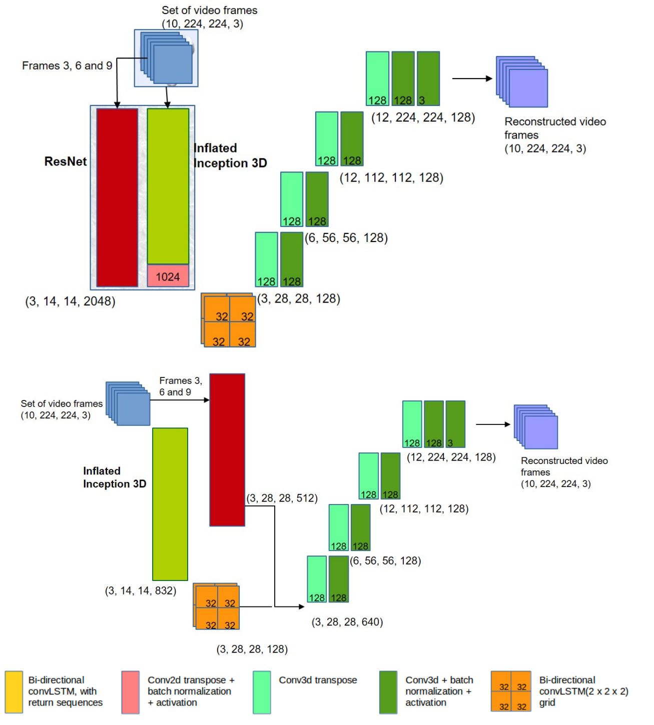

Fig. 3. Architecture (N_3) using the two configurations explained in Section III-B3

4) Network with ResNet as the base and skip connections: The above three architectures were variations of ideas from image processing for detection and classification. Although the temporal information is being captured in the form of 3D convolution units and LSTM units, reconstruction in videos requires a better approach. The final architecture makes use of ResNet for spatial feature extraction and LSTM for capturing the spatio-temporal variation. During the reconstruction phase, intermediate ResNet outputs are introduced as skip connections in deconvolution layers. The proposed network is shown in Figure 4.

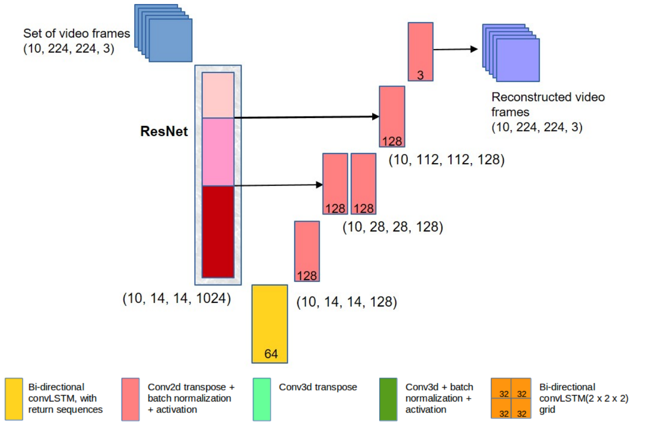

Fig. 4. Architecture (N_4) using ResNet and skip connections

C. Experimental analysis and observations for video reconstruction

Something-Something-V2 action recognition dataset is used for training our video reconstruction networks, and the ability of effective feature extraction is analysed by looking at the quality of reconstruction. This dataset contains over five lakh videos of mainly humans performing basic actions from 174 action classes. The action classes in this dataset are highly challenging since they involve atomic actions such as pushing-pulling (moving left to right vs moving right to left of the frame), picking and placing (moving top to bottom vs moving bottom to the top of the frame). Without temporal information, correct reconstruction is not possible. For all our experiments, we use a training set of ten thousand videos from this dataset, with a training, validation and test ratio of $0.8 : 0.1 : 0.1$. We use an Adam optimizer and a mean square error (MSE) loss to train the networks for one thousand epochs. As there are no benchmark results for video reconstruction, we resort to a comparison between various methods and analyse the net effect of various networks. Further, due to very poor reconstruction with some networks, we rely on qualitative rather than on quantitative comparison except for N_4 architecture.

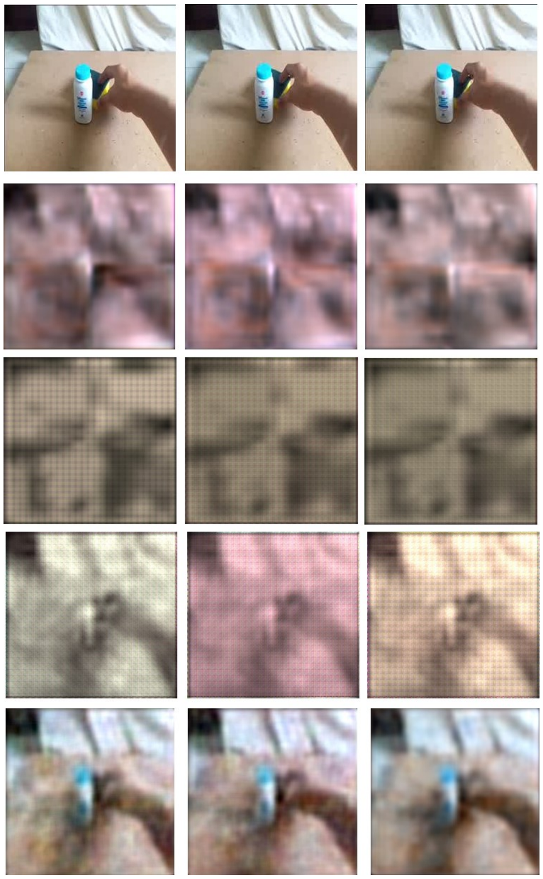

Fig. 5. Video reconstruction results using N_1 (row 2), N_2 (row 3), N_{3A} (row 4) and N_{3B} (row 5); row 1 shows the input image.

The reconstruction results of N_1, N_2, N_{3A} and N_{3B} are shown in Fig. 5. The I3D based network N_1 resulted in very poor reconstruction as shown in the second row of Fig. 5. ResNet based reconstruction architecture N_2 is able to capture information about the action objects, which is critical for many video-based object detection and classification problems. However, the RGB information and temporal variation are not reconstructed appropriately (third row of Fig. 5). Relative to the first two architectures, N_{3A} and N_{3B} showed better performance due to the combination of convolution and LSTM features being used. Specifically, combining ResNet and LSTM outputs result in better RGB reconstruction (last row of Fig. 5). The reconstruction results for N_4 are shown in Fig. 6. The frame-wise ResNet encoder captures spatial features, and LSTM takes the individual frame output at each timestep and captures the spatio-temporal representation. Along with skip connections from ResNet intermediate outputs, this gives a good reconstruction performance as shown in the middle row of Fig. 6. To assess the effect of skip connections in reconstruction, the last skip connection in N_4 was removed and the network was trained again. The last row of Fig. 6 shows the deterioration in results for this case. For quantitative comparison, Structural Similarity Index (SSIM), Peak Signal to Noise Ratio (PSNR) and Mean Square Error (MSE) between input and output of N_4 as well as between input and output of N_4 without the last skip connection are tabulated in Table I. With all the skip connections, the metric values are found to be better demonstrating the usefulness of skip connections in reconstructing spatial properties in the video.

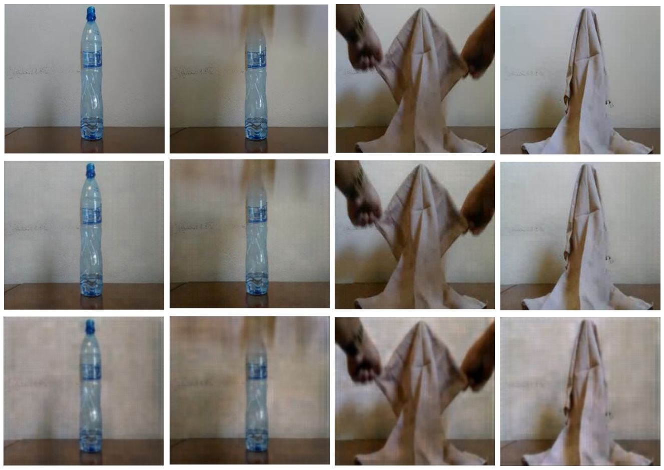

Fig. 6. Video reconstruction using network N_4: row 1 - input; row 2 - output; row 3 - output with a single skip connection

TABLE I
QUANTITATIVE EVALUATION OF TWO CONFIGURATIONS OF N_4

Metric	N_4	N_4
with one		skip connection
SSIM	0.915	0.831
PSNR	30.06	27.63
MSE	44.55	87.8

From the experimental analysis of different deep learning architectures for video reconstruction, we make observations about each block and its ability to capture spatio-temporal information from the video frames.

1) ResNet features capture spatial features extremely well as is already seen in many image recognition applications. It plays a major role in the video reconstruction. It is an important component in video applications that require frame-level outputs such as video frame prediction and video segmentation.

2) I3D model shows the ability to capture spatio-temporal variations, but is unable to get back to the original frames on its own. Further, the temporal information captured is limited to the short-term often smaller than the actions defined in any datasets.

3) The role of convLSTM is analysed, especially in the context of the N4 architecture, where the network gives the best video reconstruction. Further analysis into intermediate outputs show that ResNet almost entirely contributes to this reconstruction. This architecture is similar to a U-net type of framework, with the spatial features alone getting captured and the frames getting reconstructed. Other works in literature also point out the ineffectiveness of convLSTM to capture temporal variations in video data [26], [27].

Based on the advantages offered by each of the components in the above architectures, we design a video object segmentation network that is pre-trained on the Something-something-V2 dataset for feature learning and fine-tuned on the DAVIS dataset.

IV. Video Object Segmentation using ST-VOS network

We construct an architecture for spatio-temporal video object segmentation (ST-VOS) network using three main components: ResNet, Inception 3D and ST-LSTM. These components have shown to individually work well in different tasks. ResNet is the popular choice as a pre-trained model for multiple image applications and our experiments above also show it to be an important component for video analysis; the I3D network has been shown to perform well on video action recognition tasks, and we observe that it is capable of short-term spatio-temporal feature capture. Based on our observation of the inability of convLSTM in capturing temporal variations, we look at the recent works proposing multiple convLSTM variations for spatio-temporal information capture [27] [26]. ST-LSTM has shown good results on video frame prediction tasks, and its internal structure enables feature retention over multiple layers. Taking a combination of these elements, we design the network as shown in Figure 7.

The input to the network is a set of ten RGB frames of size (224, 224, 3). This is passed through the I3D model, and an intermediate output of size (5, 28, 28, 64) is extracted. This is input to an ST-LSTM sub-network, consisting of four layers with filter sizes as shown in Fig. 7. We tap all the intermediate outputs and then feed it to the deconvolution sub-network consisting of 3D convolution layers and 3D transpose convolution layers. The input is parallelly passed through ResNet and the outputs at multiple levels are captured. This is input to the corresponding levels in the deconvolution block to compute the final object segmentation map.

The proposed network is validated in two different ways in line with the literature using DAVIS dataset released in 2009.

Zero-Shot VOS: For the unsupervised video object segmentation, we train the network on the binary object segmentation mask of a set of videos S_{vid} and evaluate it on unseen test videos (new scenes) containing both similar objects and new objects. We pre-train the network on the Something-something V2 dataset for video reconstruction and fine-tune the last four layers for video object segmentation.

One-Shot VOS: We extend the same network for the task of one-shot video object segmentation by performing offline training. We incorporate the object segmentation mask of the first frame into the network by feeding it along with the ResNet features to the deconvolution sub-network. We resize the segmentation mask to match the ResNet intermediate outputs tapped at two levels. At each scale, these outputs are concatenated with the deconvolution layer outputs and input to the next layer. We train this network on the same set of videos S_{vid} and evaluate on the test videos.

A. Experiments and Results

In this section, details about the dataset used, experimental details and the results are presented. Due to complexity in video processing, a combination of semi-supervised and unsupervised method needs to be validated. Further, the effect of multiple network sub-blocks needs to be assessed and this is achieved using the ablation study.

Dataset: The DAVIS challenge [8] features a standard track on semi-supervised video object segmentation, in which the mask of the first frame is provided, and the segmentation for the subsequent frames should be performed. The DAVIS dataset [8] is a set of densely annotated video scenes for video object segmentation. The recent DAVIS challenge introduces a new track on unsupervised VOS or zero-shot VOS, in which, given a new set of video frames, the network should be able to segment the objects of interest from all the frames, without any prior knowledge about the objects in it. We do not train for identifying each object separately, but only train to generate the binary segmentation map for each frame, where each pixel indicates the probability of being an object of interest. We use the DAVIS 2017 dataset that consists of 60 training video sequences and 30 testing video sequences. Every frame is annotated with one or more objects. The video sequences contain an average of 70 frames. Most of the video sequences are of resolution 3840×2160, but we have used the standard down-sampled version of 480p resolution (720×480) in all our experiments.

Training details: We train according to the network configuration shown in Fig. 7 taking ten frames at a time. From the training set of 60 video sequences, we generate 600 ten-frame samples to form the training set. Recurrent units are known to underperform during training due to multiple issues and the following tweaks were made to overcome: in the deconvolution sub-network, each [conv 3D transpose +

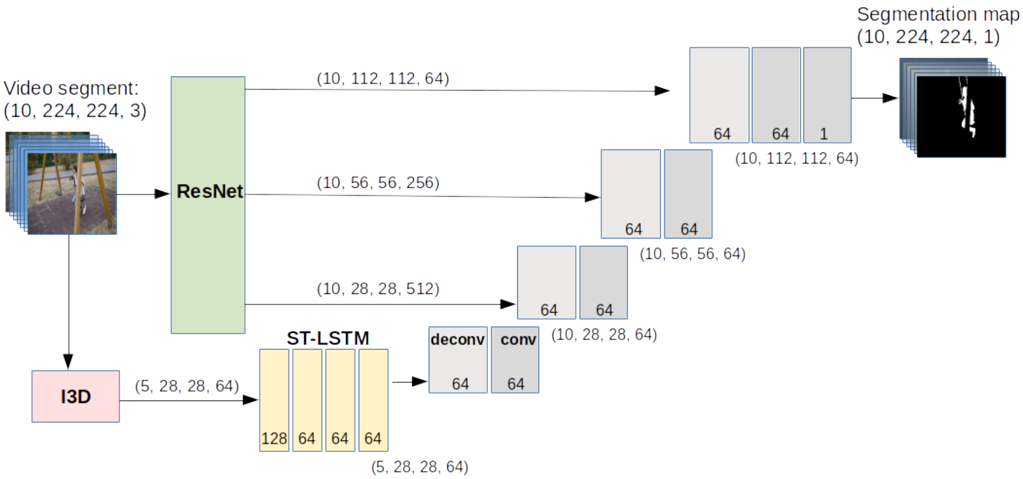

Fig. 7. Proposed network architecture for video object segmentation.

conv 3D] layer is followed by a dropout layer (prob=0.2); a batch normalization layer and a leaky ReLU activation layer was added, except the last layer which has sigmoid activation. In addition to this, Gaussian noise was added between the deconvolution blocks. The loss is set to binary cross-entropy and Adam optimizer is used. A constant learning rate of 0.001 is maintained up to 5000 iterations after which it is reduced by half every 1000 iterations. All the networks are trained for 7000 iterations. All the weights of the pre-trained I3D model and the ResNet model are frozen during training.

Zero Shot VOS and Ablation study: An ablation study was conducted to look at the effect of removing a network component on the model performance. We try three variations of the model: 1) ST-VOS without I3D; 2) ST-VOS without ST-LSTM; 3) the complete network ST-VOS. We have not considered ST-VOS without ResNet skip connections for evaluation since there is a significant dip in performance due to the absence of spatial details captured by ResNet as concluded earlier. Figure 9 shows the output of these models on test video samples for zero-shot VOS. We can see from the visual outputs that the segmentation becomes better with the addition of 3D convolution networks. The I3D features show great capability in capturing spatio-temporal features and there is only slight improvement in the performance after adding ST-LSTM layers to I3D features. This can be seen in the segmentation outputs, especially in reducing false regions, and better segmentations at the object boundaries, leading to a higher mean boundary F-score.

One-Shot VOS: For one-shot VOS, the mask of the first frame is concatenated with the ResNet skip connections at different scales, and the network is trained with an Adam optimizer and a cross-entropy loss computed over the rest of the nine frames. Figure 10 shows the visual outputs of one-shot VOS for five challenging scenarios: static objects in the background, object scale variation, multiple object instances, occlusion and cluttered background.

Quantitative Evaluation: For quantitative evaluation of the network variations, we compute the following standard metrics: Mean Jaccard index (J) and Mean boundary F-score (F). We compare the performance of our network for both zero-shot VOS and one-shot VOS, with other approaches: OSVOS [15], RVOS [24] and a self-supervised approach using pixel correspondence matching (referred to as CorrFlow) [13]. Table II summarises this evaluation. The quantitative results of the proposed network for one-shot VOS shows comparable performance with online training method OSVOS and does much better than the state-of-art self-supervised approach. Using ST-LSTM instead of convLSTM boosts performance as seen by the performance of RVOS. From the visual outputs in difficult scenes, we can infer that the network captures object features effectively that allows segmentation consistency for different types of scenes.

One-Shot VOS with noisy labels: To evaluate the robustness of features captured by one-shot VOS network, we provide the network with a noisy object mask to the first frame. To do this, we modify the object annotations in two ways: we extract different size object bounding boxes from the pixel annotation; we perform small random translations on the extracted bounding box masks. We train the network on the same video set S_{vos}, but with these modified annotations and observe the deterioration in performance with an increase in noise. Figure 8 plots the change in J and F with the increase in wrong labels. From the graph in figure 8, it can be clearly seen that the deterioration in performance is quite low even with a sufficiently large number of noisy pixels. Assuming that on an average, an object occupies one-fifth of an image and considering an image size of (224 × 224), 1200 noisy pixels correspond to about 12% of the object. It can be inferred that the performance of the network, given a mask with 12% noisy pixels is as good as that given an accurate object mask. This shows great promise for applications that require unsupervised or semi-supervised annotation.

TABLE II
TABLE SHOWING PERFORMANCE METRICS – MEAN JACCARD INDEX (J), MEAN BOUNDARY F-MEASURE (F) FOR COMPARISON WITH OTHER APPROACHES FOR ZERO-SHOT VOS AND ONE-SHOT VOS

Method	Zero-Shot		One-Shot	
Metric	J	F	J	F
RVOS	23.0	29.9	48.0	52.6
CorrFlow	-	-	48.4	52.2
OSVOS	-	-	56.6	63.9
ST-VOS without I3D	31.6	34.2	-	-
ST-VOS without ST-LSTM	42.9	43.9	-	-
ST-VOS	43.2	44.7	52.9	60.4

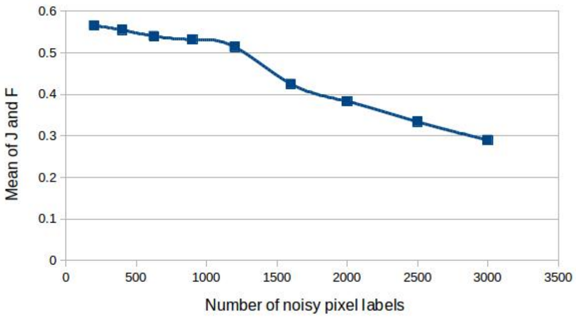

Fig. 8. Graph showing the varation of the mean of J and F for one-shot VOS, with increase in noisy label pixels in the first frame segmentation mask

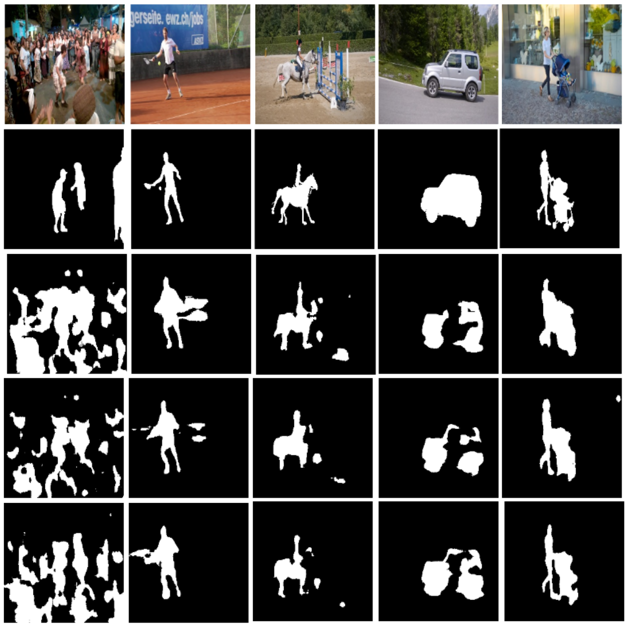

Fig. 9. Visual outputs of testing our network variations for video object segmentation on DAVIS dataset; Row 1: Input image, Row 2: Segmentation groundtruth; Row 3: ResNet-STLSTM output, Row 4: I3D-ResNet output, Row 5: I3D-STLSTM-ResNet output

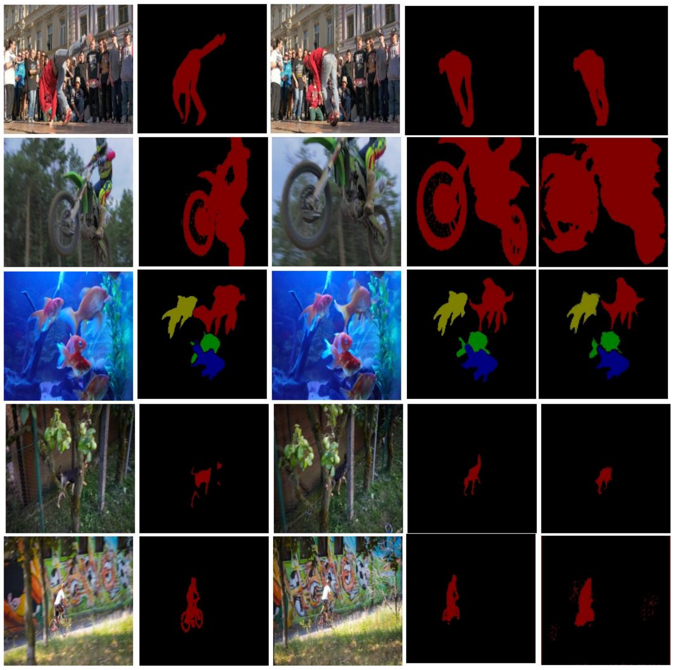

Fig. 10. Visual outputs of testing our network for one shot video object segmentation on DAVIS dataset: Rows: 1) First frame, 2) Object mask, 3) Test frame, 4) Ground truth mask, 5) Output of our network; five difficult scenarios considered (from top to bottom): static objects in background, object scale variation, multiple object instances, occlusion and cluttered background.

Time complexity: For video object segmentation of frames of size (224, 224, 3), the proposed ST-VOS network takes 56 ms per frame and the network without ST-LSTM layers takes 48 ms per frame on a Tesla K80 GPU. The time taken for training each network for 7000 iterations for a training set of 600 video segments with ten frames each is about 15 hours.

V. CONCLUSION

In this work, we formulate the objective of video feature extraction as a video reconstruction problem. We explore multiple architectures using different deep learning blocks and assess the ability of the blocks in capturing spatio-temporal features effectively. The effect of LSTM, I3D and ResNet are discussed in detail with respect to the video representation problem. For video reconstruction, an SSIM value of 0.915 is obtained on Something-Something-v2 dataset. We further design a spatio-temporal video object segmentation network based on the reconstruction results obtained earlier. We successfully implement the network for the application of zero-shot VOS and one-shot VOS on the DAVIS 2017 dataset. We obtain a mean Jaccard index of 43.2 and 52.9 for zero-shot and one-shot VOS respectively. F-measure of 44.7 and 60.4 respectively is obtained for zero-shot and one-shot VOS respectively. Further evaluation in the presence of noisy labels shows the robustness of our network in performing accurate video object segmentation.

REFERENCES

[1] D.-A. Huang, V. Ramanathan, D. Mahajan, L. Torresani, M. Paluri, L. Fei-Fei, and J. C. Niebles, “What makes a video a video: Analyzing temporal information in video understanding models and datasets,” *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7366–7375, 2018.

[2] K. Soomro, A. R. Zamir, and M. Shah, “UCF101: A dataset of 101 human actions classes from videos in the wild,” *CoRR*, vol. abs/1212.0402, 2012. [Online]. Available: http://arxiv.org/abs/1212.0402

[3] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, M. Suleyman, and A. Zisserman, “The kinetics human action video dataset,” *CoRR*, vol. abs/1705.06950, 2017. [Online]. Available: http://arxiv.org/abs/1705.06950

[4] R. Goyal, S. E. Kahou, V. Michalski, J. Materzynska, S. Westphal, H. Kim, V. Haenel, I. Fründ, P. Yianilos, M. Mueller-Freitag, F. Hoppe, C. Thurau, I. Bax, and R. Memisevic, “The ”something something” video database for learning and evaluating visual common sense,” *CoRR*, vol. abs/1706.04261, 2017. [Online]. Available: http://arxiv.org/abs/1706.04261

[5] “The 20bn-jester dataset v1,” https://20bn.com/datasets/jester.

[6] B. Zhou, A. Andonian, and A. Torralba, “Temporal relational reasoning in videos,” *CoRR*, vol. abs/1711.08496, 2017. [Online]. Available: http://arxiv.org/abs/1711.08496

[7] O. Köpüklü, N. Köse, and G. Rigoll, “Motion fused frames: Data level fusion strategy for hand gesture recognition,” *CoRR*, vol. abs/1804.07187, 2018. [Online]. Available: http://arxiv.org/abs/1804.07187

[8] J. Pont-Tuset, F. Perazzi, S. Caelles, P. Arbeláez, A. Sorkine-Hornung, and L. Van Gool, “The 2017 davis challenge on video object segmentation,” *arXiv:1704.00675*, 2017.

[9] B. Fernando, H. Bilen, E. Gavves, and S. Gould, “Self-supervised video representation learning with odd-one-out networks,” *CoRR*, vol. abs/1611.06646, 2016. [Online]. Available: http://arxiv.org/abs/1611.06646

[10] H.-Y. Lee, J.-B. Huang, M. Singh, and M.-H. Yang, “Unsupervised representation learning by sorting sequences,” 10 2017, pp. 667–676.

[11] I. Misra, C. L. Zitnick, and M. Hebert, “Unsupervised learning using sequential verification for action recognition,” *CoRR*, vol. abs/1603.08561, 2016. [Online]. Available: http://arxiv.org/abs/1603.08561

[12] L. Jing and Y. Tian, “Self-supervised spatiotemporal feature learning by video geometric transformations,” *CoRR*, vol. abs/1811.11387, 2018. [Online]. Available: http://arxiv.org/abs/1811.11387

[13] Z. Lai and W. Xie, “Self-supervised learning for video correspondence flow,” *CoRR*, vol. abs/1905.00875, 2019. [Online]. Available: http://arxiv.org/abs/1905.00875

[14] P. Liu, M. R. Lyu, I. King, and J. Xu, “Selflow: Self-supervised learning of optical flow,” *CoRR*, vol. abs/1904.09117, 2019. [Online]. Available: http://arxiv.org/abs/1904.09117

[15] S. Caelles, K. Maninis, J. Pont-Tuset, L. Leal-Taixé, D. Cremers, and L. V. Gool, “One-shot video object segmentation,” *CoRR*, vol. abs/1611.05198, 2016. [Online]. Available: http://arxiv.org/abs/1611.05198

[16] A. Khoreva, F. Perazzi, R. Benenson, B. Schiele, and A. Sorkine-Hornung, “Learning video object segmentation from static images,” *CoRR*, vol. abs/1612.02646, 2016. [Online]. Available: http://arxiv.org/abs/1612.02646

[17] H. Liu and J. Jiang, “U-net based multi-instance video object segmentation,” *CoRR*, vol. abs/1905.07826, 2019. [Online]. Available: http://arxiv.org/abs/1905.07826

[18] A. Shaban, A. Firl, A. Humayun, J. Yuan, X. Wang, P. Lei, N. Dhanda, B. Boots, J. M. Rehg, and F. Li, “Multiple-instance video segmentation with sequence-specific object proposals,” *The 2017 DAVIS Challenge on Video Object Segmentation - CVPR Workshops*, 2017.

[19] X. Li, Y. Qi, Z. Wang, K. Chen, Z. Liu, J. Shi, P. Luo, X. Tang, and C. C. Loy, “Video object segmentation with re-identification,” *CoRR*, vol. abs/1708.00197, 2017. [Online]. Available: http://arxiv.org/abs/1708.00197

[20] S. Caelles, J. Pont-Tuset, F. Perazzi, A. Montes, K. Maninis, and L. V. Gool, “The 2019 DAVIS challenge on VOS: unsupervised multi-object segmentation,” *CoRR*, vol. abs/1905.00737, 2019. [Online]. Available: http://arxiv.org/abs/1905.00737

[21] W. Wang, H. Song, S. Zhao, J. Shen, S. Zhao, S. C. H. Hoi, and H. Ling, “Learning unsupervised video object segmentation through visual attention,” in *CVPR*, 2019.

[22] X. Lu, W. Wang, C. Ma, J. Shen, L. Shao, and F. Porikli, “See more, know more: Unsupervised video object segmentation with co-attention siamese networks,” in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[23] S. Li, B. Seybold, A. Vorobyov, X. Lei, and C.-C. J. Kuo, “Unsupervised video object segmentation with motion-based bilateral networks,” in *Computer Vision – ECCV 2018*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds. Cham: Springer International Publishing, 2018, pp. 215–231.

[24] C. Ventura, M. Bellver, A. Girbau, A. Salvador, F. Marqués, and X. Giró i Nieto, “RVOS: end-to-end recurrent network for video object segmentation,” *CoRR*, vol. abs/1903.05612, 2019. [Online]. Available: http://arxiv.org/abs/1903.05612

[25] J. Carreira and A. Zisserman, “Quo vadis, action recognition? A new model and the kinetics dataset,” *CoRR*, vol. abs/1705.07750, 2017. [Online]. Available: http://arxiv.org/abs/1705.07750

[26] Y. Wang, M. Long, J. Wang, Z. Gao, and P. S. Yu, “Predrnn: Recurrent neural networks for predictive learning using spatiotemporal lstms,” in *Advances in Neural Information Processing Systems 30*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds. Curran Associates, Inc., 2017, pp. 879–888. [Online]. Available: http://papers.nips.cc/paper/6689-predrnn-recurrent-neural-networks-for-predictive-learning-using-spatiotemporal-lstms.pdf

[27] Y. Wang, L. Jiang, M. hsuan Yang, J. Li, M. Long, and F.-F. Li, “Eidetic 3d lstm: A model for video prediction and beyond,” in *ICLR*, 2019.