UVid-Net: Enhanced Semantic Segmentation of UAV Aerial Videos by Embedding Temporal Information

Girisha S., *Member, IEEE*, Ujjwal Verma, *Senior Member, IEEE*, Manohara Pai M. M., *Senior Member, IEEE*, and Radhika M. Pai, *Senior Member, IEEE*

***Abstract*—Semantic segmentation of aerial videos has been extensively used for decision making in monitoring environmental changes, urban planning, and disaster management. The reliability of these decision support systems is dependent on the accuracy of the video semantic segmentation algorithms. The existing CNN-based video semantic segmentation methods have enhanced the image semantic segmentation methods by incorporating an additional module such as LSTM or optical flow for computing temporal dynamics of the video which is a computational overhead. The proposed research work modifies the CNN architecture by incorporating temporal information to improve the efficiency of video semantic segmentation. In this work, an enhanced encoder–decoder based CNN architecture (UVid-Net) is proposed for unmanned aerial vehicle (UAV) video semantic segmentation. The encoder of the proposed architecture embeds temporal information for temporally consistent labeling. The decoder is enhanced by introducing the feature-refiner module, which aids in accurate localization of the class labels. The proposed UVid-Net architecture for UAV video semantic segmentation is quantitatively evaluated on extended ManipalUAVid dataset. The performance metric mean Intersection over Union of 0.79 has been observed which is significantly greater than the other state-of-the-art algorithms. Further, the proposed work produced promising results even for the pretrained model of UVid-Net on urban street scene by fine tuning the final layer on UAV aerial videos.**

***Index Terms*—Deep learning, semantic segmentation, transfer learning, U-Net, unmanned aerial vehicle (UAV) video.**

I. Introduction

THE analysis of data collected from airborne sensors such as aerial images/videos is increasingly becoming a vital factor in many applications such as scene understanding, studying the ecological variations [35], [36], [44], tracking of vehicles/animals/humans [9], [25], [34], surveying the urban development [46], [53], [60], etc. Besides, aerial image analysis has been used for assessing the damage immediately after a natural disaster [17]. Typically, the aerial images are captured by different imaging modalities such as synthetic aperture radar [54] and hyper-spectral imaging [41] which are present on-board a satellite. Recently, the unmanned aerial vehicles (UAVs) have also been widely used for various applications such as disaster management, urban planning, tracking of wildlife, agricultural planning, etc. [3], [4]. Due to rapid deployment and a customized flight path, the UAV images/videos could provide additional finer details and complement satellite-based image analysis approaches for critical applications such as disaster response [31]. Besides, the UAV images could be utilized along with satellite images for better urban planning or geographical information updating. Typically, the UAV image/video analysis is limited for object detection [12], [24] and recognition [50] tasks such as building detection, road segmentation, etc. However, to the best of our knowledge, there are limited works on semantic segmentation of UAV images or videos [14], [52].

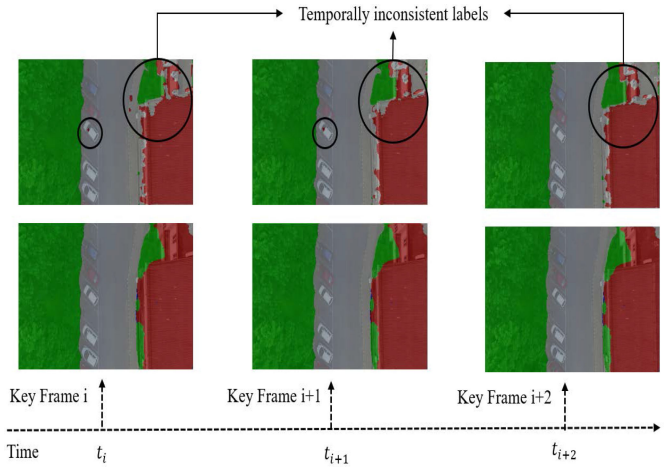


Fig. 1. Importance of temporal consistency in video scene labeling. First row represents the temporal inconsistent labels produced by image segmentation algorithm (U-Net) on videos. Second row depicts an example of temporally consistent labeling obtained using the proposed approach (UVid-Net).

Segmentation is a crucial task for scene understanding and has been used for various applications [10], [42], [47]. Semantic segmentation is a process of assigning predetermined class labels to all the pixels in an image. Semantic segmentation of an *image* is a widely studied topic in computer vision. However, the extension of semantic segmentation for video applications is a nontrivial task. One of the challenges in video semantic segmentation is to find a way to incorporate temporal information. Fig. 1 illustrates the importance of temporal information in the context of video acquired by UAV. The poor segmentation in the

Manuscript received November 27, 2020; revised January 27, 2021 and March 13, 2021; accepted March 27, 2021. Date of publication March 31, 2021; date of current version April 26, 2021. *(Girisha S. and Ujjwal Verma, contributed equally to this work.) (Corresponding author: Manohara Pai.)*

Girisha S., Manohara Pai M. M., and Radhika M. Pai are with the Department of Information and Communication Technology, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal 576104, India (e-mail: girisha3893@gmail.com; mmm.pai@manipal.edu; radhika.pai@manipal.edu).

Ujjwal Verma is with the Department of Electronics and Communication Engineering, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal 576104, India (e-mail: ujjwal.verma@manipal.edu).

Digital Object Identifier 10.1109/JSTARS.2021.3069909

greenery class can be observed in the $(i+1)$th keyframe which can be improved by embedding temporal information from the past frames.

In a typical video semantic segmentation approach, a sequential model is added on top of the framewise semantic segmentation module, thus creating an overhead [13]. Besides, features/label propagation [33], which reutilizes features/labels from previous frames has also been utilized to capture the temporal information. However, these methods depend on the establishment of pixel correspondence between two frames. Recently, video prediction based approach [61] has been used to generate new training images and has achieved state-of-the-art performance for video semantic segmentation. However, this approach uses an additional video prediction model to learn the motion information.

This work focuses on semantic segmentation of videos acquired using UAV. The proposed method demonstrates that a simple modification in the encoder branch of CNN is able to capture the temporal information from the video, thus eliminating the need for an extra sequential model for computing correspondence for feature/label propagation.

A new encoder–decoder based CNN architecture (UVid-Net) proposed in this work has two parallel branches of CNN layers for feature extraction. This new encoding path captures the temporal dynamics of the video by extracting features from multiple frames. These features are further processed by the decoder for the estimation of class labels. The proposed algorithm utilizes a new decoding path that retains the features of encoder layers for decoders. The contribution of the article can be summarized as follows.

1) The dependence of existing methods on optical flow/ConvLSTM for the establishment of temporal correspondence is an overhead for video semantic segmentation. Hence, a new encoding path is presented consisting of two parallel branches for extracting temporal and spatial features for video semantic segmentation. This new encoding path eliminates the need for an extra sequential module (ConvLSTM) or computation of optical flow for establishing temporal correspondence.
2) A modified upsampling path is proposed which uses a feature-refiner module to capture fine-grain features for accurate classification of class boundary pixels. The feature-refiner module also reduces the number of parameters (11.68% reduction) and the computational complexity (11% reduction) of the model as compared to the traditional decoder module.
3) An extended version of UAV video semantic segmentation dataset is presented. This dataset is an extension of ManipalUAVid dataset [15] and contains additional videos captured at new locations. Fine pixel-level annotations are provided for four background classes namely greenery, roads, constructions, and water bodies as per the policy adopted in [15]. The dataset is available for download at https://github.com/uverma/ManipalUAVid
4) This work also studies the performance of the proposed UVid-Net trained on a larger urban street scene dataset for semantic segmentation for segmentation of UAV aerial videos. The capability of UVid-Net to utilize transferable features allows the model to be retrained with a few labeled data.

This article is organized as follows. Section II summarizes the recent developments in video semantic segmentation. Section III describes the architecture of the proposed network UVid-Net and Section IV presents the various results obtained.

II. Related Works

Video semantic segmentation is generally addressed by utilizing traditional energy-based algorithms such as conditional random field (CRF) or deep learning based algorithms such as CNN, RNN, LSTM, etc. One of the challenges in video semantic segmentation is to embed temporal information. Learning the dynamics of the video aids in improving the performance of video semantic segmentation by ensuring temporal consistency. Despite this interest, previous works such as [6], [14], and [15] extended the traditional image semantic segmentation approach for video semantic segmentation. These approaches segment all the frames independently of each other which fails to capture the dynamics of the video. Recent advances in video semantic segmentation by utilizing spatio-temporal information can be categorized into roughly two groups: deep learning based methods and CRF-based methods.

There exist several CNN-based semantic segmentation approaches in literature such as [20], [38], [45], [57], [61], etc. The authors of [57] proposed bilateral segmentation network (BiSeNet) to capture spatial and contextual information for semantic segmentation. In a separate study [29], the authors used part-object relationship for a robust salient object segmentation. In [28], authors used multiple ASPP module to increase the density of sampling distribution. However, these approaches analyze single image, while the proposed work aims to incorporate temporal information for video semantic segmentation.

Popular CNN-based algorithms like [30] and [42] used encoder- and decoder-based architecture for learning the various patterns of the data and localizing the class labels. These algorithms are dependent on a large densely annotated dataset. However, obtaining a finely annotated large dataset is expensive, time-consuming, and challenging. To address the issue of limited training data, GANs were utilized [45]. Few authors [1], [22] used GAN to learn the dynamics of the video and perform video scene parsing. GAN can be trained to parse future frames as well as label images as proposed by [22].

Besides, temporal dynamics are also learnt using a sequential model like LSTM [52]. Moreover, LSTM is also used to select keyframes for video scene parsing [32]. Wang *et al.* [48], proposed noisy-LSTM which uses ConvLSTM for video semantic segmentation. The strategy used is to train network with noisy images to disrupt the temporal information. Recently, memory modules are also explored for learning the temporal dynamics of the video [37]. Few authors explored the attention mechanism with CNN to perform video semantic segmentation [27], [49].

Wang *et al.* [49] proposed TMANet which uses attention mechanism to capture long-range temporal information required for video semantic segmentation. In another study [55], authors used pixel-level matching between two consecutive frames to obtain global and local similarity maps for video object segmentation. However, it is challenging to determine the attention coefficients. Optical flow is another popular choice for the establishment of temporal correspondence between two consecutive frames [21]. Few studies such as [38] and[61] proposed to predict labels and images jointly to efficiently train deep learning models with less training data. However, the dependence of deep learning algorithms on large annotated datasets limits the development of deep learning algorithms for other contexts such as UAV, etc.

Many researchers have explored CRF for incorporating spatio-temporal information in video semantic segmentation. CRF is a graphical model that captures a large spatial relationship between pixels. Hence, it is widely used in literature for context-aware scene parsing [8], [43]. CRF can be extended to incorporate temporal information as shown in few literatures such as [2], [5], [8], [23], [26], and [59], but it depends on the reliable estimation of temporal links. The authors of [26] utilized 3-D CRF along with optimized feature space for video semantic segmentation. However, 3-D CRF is impractical for videos since it is computationally expensive. Potential energies based on temporal information were also explored for producing temporally consistent labels [5]. Few researchers incorporated CNN within CRF frame work to obtain initial estimation of class labels [2], [59]. Authors of [23] used conditional restricted Boltzmann machine along with CRF to learn the temporal and shape features required for video semantic segmentation. In general, optical flow is widely used to establish the temporal link and propagate features and labels. However, estimation of accurate optical flow is an overhead for real-time video semantic segmentation. In the recent work of [16], a new potential term was proposed to enhance the temporal smoothness of video semantic segmentation without the usage of optical flow. In another study, higher order potential energies were explored for video semantic segmentation [8]. Class labels in CRF are inferred by using an inference algorithm which is computationally intensive and impractical for video processing.

The existing state-of-the-art method for video semantic segmentation predicts frames and its labels from the historic data [61]. However, this approach is dependent on a reliable estimation of temporal correspondence between two consecutive frames. Temporal links are generally established by utilizing dense optical flow-based methods [26]. Optical flow estimation is an overhead, and the accuracy of semantic segmentation depends on the accuracy of optical flow estimation. Besides, the error in optical flow estimation can lead to misaligned predicted labels in the future frames, thus affecting the accuracy of the segmentation. The proposed work eliminates the need for computing optical flow, thus reducing the overhead.

In this work, a two-branch encoder is proposed for incorporating temporal smoothness in video semantic segmentation. Multibranch CNNs are popularly used in video processing due to their ability to capture the relationship between the sequence of frames. Several authors used multibranch CNNs to perform video classification [51], action recognition [39], and video captioning [58]. Few authors utilized multibranch CNN architecture to provide attention mechanism. Authors of [40] explored multibranch CNN to extract features from different frames. In [19], the authors proposed to utilize multiple shallow networks to extract features from consecutive frames to perform video semantic segmentation. To the best of our knowledge, lightweight multibranch CNNs are not explored to perform video semantic segmentation of UAV videos.

III. Methodology

This section describes the encoder (Section III-A) and decoder module (Section III-B) of the proposed approach. Figs. 2 and 3 show the proposed architecture with U-Net and ResNet-50 feature extractor, respectively. In a typical video, the changes between two consecutive frames are very minimum, and, hence, processing every frame is redundant and time-consuming for video semantic segmentation. However, selecting keyframes at constant interval may result in loss of useful information required for temporal consistency. This would be detrimental for video semantic segmentation methods which depend on temporal features. Hence, in the present study, the keyframes are identified using the shot boundary detection approach presented in [15] (on an average, a shot consists of 15–20 frames). The use of shot boundary detection method for dynamically identifying the keyframes ensures that the frames containing useful information are not ignored.

Let us represent the ith frame from the lth shot in a video as f_i^l. The inputs to the two branches of UVid-Net (Figs. 2 and 3) are the two frames from two consecutive shots: $f_{(n/2+1)}^{(l-1)}$ (upper branch) and $f_{n/2}^{l}$ (lower branch), where n represents the total number of frames in a shot. These two frames correspond to the next frame after the middle frame of the previous shot $f_{(n/2+1)}^{(l-1)}$ and the middle frame from the current shot. These two input frames produce the semantic segmentation for the middle frame of the current shot $f_{n/2}^{l}$. For the first shot, since there is no prior shot, the first frame (f_1^1) of the video and middle frame ($f_{n/2}^1$) of the first shot is considered as input to the network. In the rest of this document, the middle frame of a shot is considered as the keyframe, as per the policy followed for UAV video semantic segmentation [15].

A. Encoder

In this work, the performance of two different architectures (U-Net and ResNet-50 encoders) is studied for feature extraction. U-Net encoder consists of a convolutional layer and maxpool layers for feature extraction. The ResNet-50 feature extractor consists of residual blocks that help in alleviating the vanishing gradient. These two feature extractors are different, and comparing their performance on multibranch CNN helps us in providing insight into the robustness of the model. In the following text, UVid-Net (U-Net encoder) and UVid-Net (ResNet-50 encoder) refer to the proposed architecture with U-Net encoder and ResNet-50 encoder module, respectively.

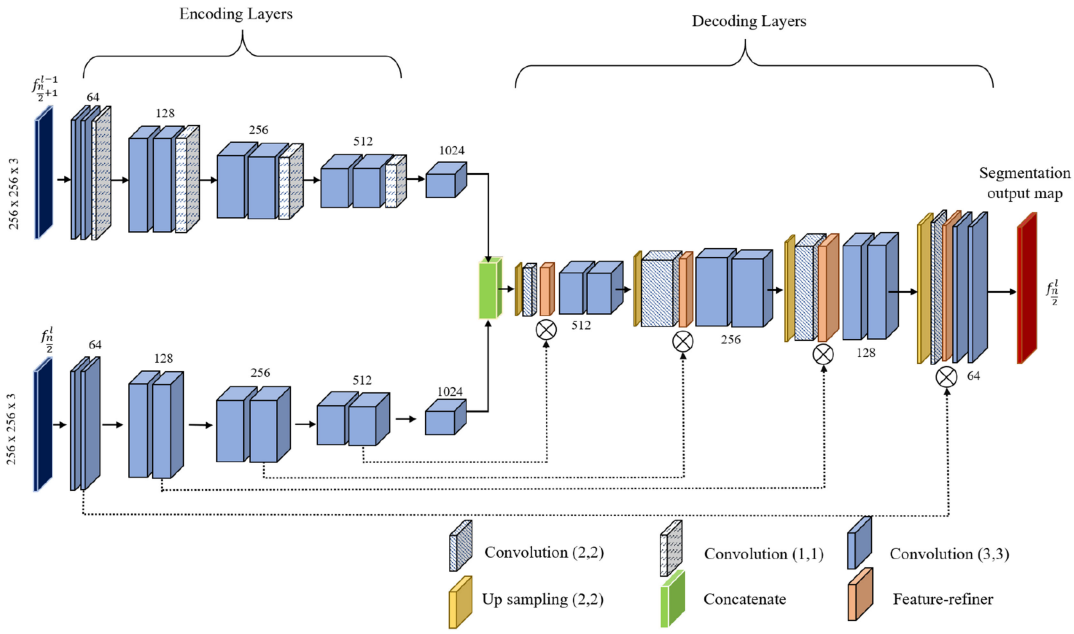


Fig. 2. UVid-Net: Overview of the proposed architecture for UAV video semantic segmentation (U-Net encoder). The architecture consists of encoding path to extract spatio-temporal features and an upsampling path which produces smoother segmentation boundaries.

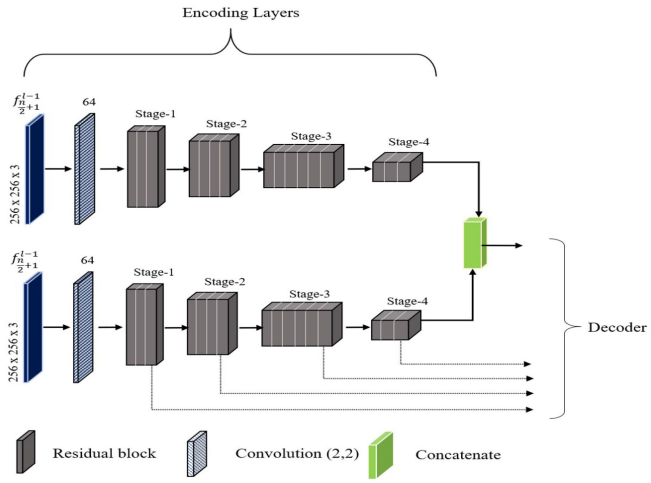


Fig. 3. UVid-Net (ResNet-50 encoder): overview of the proposed architecture for UAV video semantic segmentation with ResNet-50 encoder.

It may be noted that the decoder module is identical for both the architectures.

1) U-Net Encoder: The upper branch of the encoder (Fig. 2) contains four blocks. Each block consists of two consecutive 3×3 convolution layers with batch normalization and ReLU activation function as in the encoder of U-Net [57]. Finally, the activation is then passed through a 1×1 convolution layer which is additionally introduced to reduce the dimensions of the feature maps. Finally, a maxpooling layer with stride (2,2) is applied to extract the most prominent features for the subsequent layers. As in the traditional U-Net, the number of feature maps doubles after each maxpooling operation, starting with 64 feature maps for the first block.

The lower branch of the encoder also consists of four blocks. Each block in the lower branch has a 3×3 convolution layer with batch normalization and ReLU activation function and the second set of 3×3 convolution layer with batch normalization and ReLU activation function. This is followed by a maxpooling layer that extracts most prominent features. Similar to the upper branch, the number of feature maps doubles after each maxpooling operation.

The features extracted by the upper and lower branches of the encoder are fed to two separate bottleneck layers consisting of 3×3 convolution with 1024 features maps. Finally, the activation of both these branches is concatenated and fed to the decoder.

2) ResNet-50 Encoder: Besides the U-Net based encoder described above, the ResNet-50 architecture (Fig. 3) could also be used as a branch in the encoder. ResNet-50 is a CNN architecture proposed for image classification. This architecture proposed the idea of skipping a few layers to learn identity mapping. ResNet-50 has also been widely used as a feature extractor for transfer learning applications [18].

In the present study, the upper branch and lower branch consist of identical ResNet50 architecture to extract features (Fig. 3). This architecture consists of an initial convolution operation with kernel size (7×7) followed by batch normalization layer and ReLU activation function. Subsequently, a maxpool operation with kernel size (3×3) is applied. Followed by the maxpool operation, the architecture consists of four stages. The first stage consists of three residual blocks, each containing three layers. Each of these residual blocks consists of 64, 64, and 128 filters. The second stage consists of four residual blocks with three layers each. These three layers use 128, 128, and 256 filters. The third stage consists of six residual blocks with three layers each. These layers use 256, 256, and 512 filters. The fourth

stage consists of three residual blocks with three layers each. These layers use 512, 512, and 1024 filters. The first residual blocks of stage 2, 3, and 4 utilize stride operation to reduce the input dimension by 2 in terms of width and height. First and last layers in every residual block consist of (1×1) kernel size and the second layer consists of (3×3) kernel size. All residual block consists of identity connection which solves the vanishing gradient problem.

The activations of upper and lower ResNet50 branches are concatenated and are further used by the decoder to perform semantic segmentation.

B. Decoder

In an encoder–decoder based architecture, the consecutive maxpooling operations in encoder reduces feature map's size and results in the loss of spatial resolution. Hence, to compensate for this loss of information, skip connections are popularly used from encoding layers to decoding layers [30], [42]. Networks like U-Net use concatenation operation where the feature maps from the last layer of each block in the encoder are stacked with the feature maps of corresponding decoding layers. Here, we argue that elementwise multiplication of the feature maps from the last layer of each block in encoder with the corresponding decoding layers results in better representation of feature maps. This module which performs elementwise multiplication of feature maps is called as feature-refiner since it **refines** the features of the corresponding encoding path. In addition to the improvement in segmentation, the proposed feature-refiner module reduces the number of learnable parameters as compared to the concatenation operation. For instance, the total number of parameters for UVid-Net (U-Net encoder) with multiplication is 23 745 032, whereas the total number of parameters for UVid-Net (U-Net encoder) with concatenation is 26 878 472. The experimental results (Section IV-C) show that the elementwise multiplication of the encoder feature map with the corresponding decoder feature map produces a more smoother segmentation map.

As discussed earlier, the decoder module is identical for both UVid-Net (U-Net encoder) and UVid-Net (ResNet-50 encoder) (Figs. 2 and 3). The decoder path of the proposed architecture contains four blocks. Each of these blocks consists of an upsampling layer with stride 2. This is followed by a convolution layer with filter size (2,2). The output of this is passed through a feature-refiner module which multiplies the corresponding feature maps of the encoder (lower branch) and the decoder. Note that the last layer of each stage/block of the lower branch encoder is merged with corresponding decoder layers. This is followed by convolution layers and the ReLU activation layer. At the last layer, the SoftMax layer is applied to obtain the probability of pixels belonging to each class.

IV. Results and Discussion

In the present study, an extended version of ManipalUAVid [15] dataset is used to evaluate the performance of the UVid-Net for UAV video semantic segmentation. The proposed architecture is trained by utilizing categorical cross-entropy loss with Adam optimizer for learning the parameters of the model. In this section, it is shown experimentally that the proposed encoder module is able to incorporate temporal smoothness for video semantic segmentation (Section IV-B). Further, the effectiveness of the feature-refiner in the decoder module is demonstrated in Section IV-C. Finally, the performance of the proposed architecture is compared with the state-of-the-art methods for video semantic segmentation (Section IV-D).

A. Dataset: ManipalUAVid

This article presents an extended version of ManipalUAVid [15] dataset for semantic segmentation of UAV videos. This extended dataset consists of new videos captured at additional locations. The extended dataset consists of 37 videos with annotations provided for 711 keyframes. The pixel-level annotations are provided for four background classes viz., greenery, construction, road, and water bodies. The videos are captured at 29 frames per second and at a resolution of 1280×720 pixels. The keyframes are identified by following the shot boundary detection approach mentioned in [15], and, on an average, a shot consists of 15–20 frames. More details of this dataset can be found in [15]. The ManipalUAVid presented in [15] contains 33 videos and annotations were provided for 667 keyframes. Besides, the performance of semantic segmentation algorithms that analyze each keyframe individually was provided in [15] on the ManipalUAVid dataset. The earlier version of ManipalUAVid dataset [15] consists of last two keyframes of each video in the test split which might not be sufficient to observe the temporal smoothness or the error (if any) accumulated over the period of time in the video. Therefore, in this work, ManipalUAVid is extended by incorporating four new videos (total key frames: 44) which are entirely in the test split. Besides, the training-test split distribution is slightly modified so that a greater number of frames (4–5 frames) per video is included in the test split of this updated dataset. This aids in evaluating the video semantic segmentation models for temporal consistency.

Following the same protocol [15], the performance of UVid-Net is evaluated by comparing the keyframes segmented using UVid-Net with the ground truth. In ManipalUAVid, middle frames of a shot ($f^l_{(n/2)}$) are considered as the keyframes. As discussed earlier, two frames ($f^{(l-1)}_{(n/2+1)}$ and $f^l_{(n/2)}$) are provided as the input to UVid-Net for semantic segmentation of $f^l_{(n/2)}$ ($l \neq 1$). The dataset is divided into train, validation and test split which consists of 569, 71, and 71 keyframes, respectively. The following metrics are computed to evaluate the performance of the proposed method: mean Intersection over Union (mIoU), precision, recall, and F1-score. It may be noted that the values of the evaluation metrics obtained in this study are different from that reported in [15] due to additional videos being added in the dataset.

B. Evaluation of Encoder

The proposed encoder part consists of two branches that extract features from two consecutive keyframes of a video simultaneously. Two variants of UVid-Net (U-Net encoder and ResNet-50 encoder) encoders are considered in this work. To

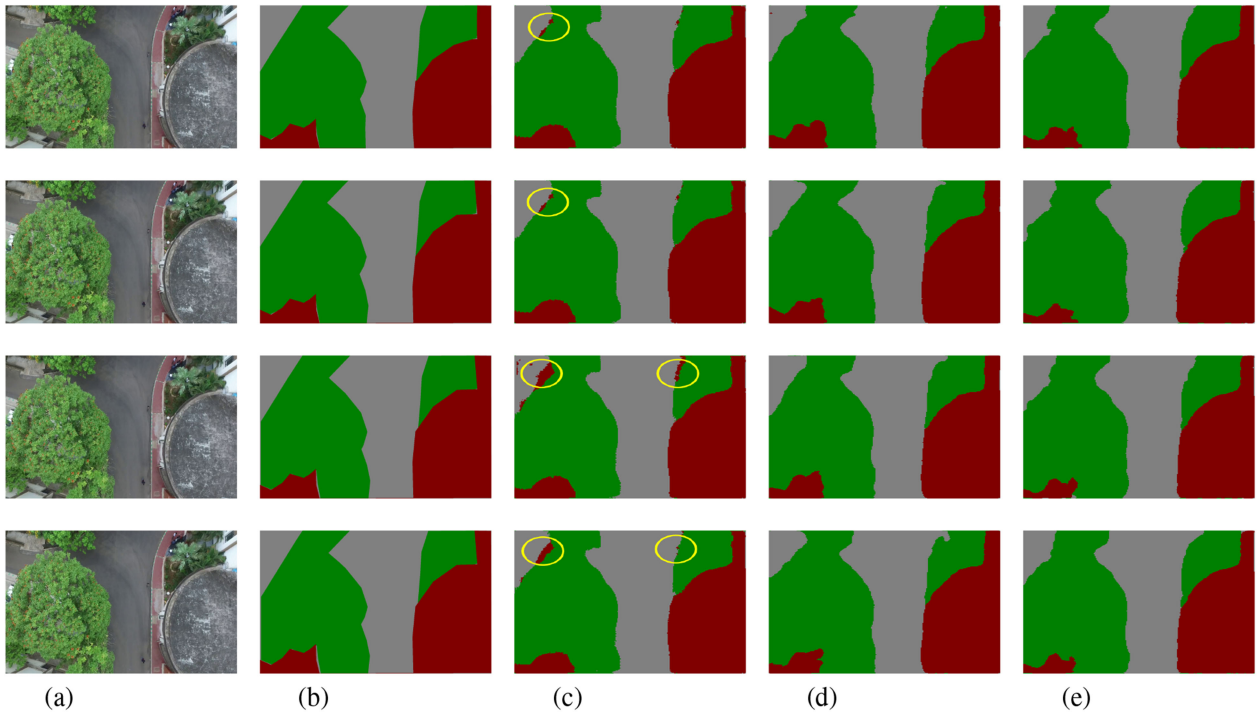


Fig. 4. Comparing the performance of the proposed two-branch encoder module with single-branch encoder. (a) Four consecutive keyframes and (b) its corresponding ground-truth images. (c) Results of single-branch encoder viz., U-Net. (d) and (e) Results of two-branch encoder architectures viz., UVid-Net (with U-Net encoder) and UVid-Net (ResNet50 encoder), respectively. Yellow circles highlight the temporal inconsistency produced by single-branch U-Net in semantic segmentation. Here, green, gray, red, and blue colors represent the greenery, road, construction, and water bodies class, respectively.

TABLE I
PERFORMANCE METRICS OF THE VARIOUS ALGORITHMS ON MANIPALUAVID DATASET

Method	Precision	Recall	F1-Score	mIoU	Learnable Parameters	FLOPs
U-Net [42]	0.89	0.89	0.89	0.75	21,593,732	62,050,187
FCN-8 [30]	0.85	0.85	0.85	0.64	134,526,856	269,028,892
DeepLabV3+ (MobileNet-V2 backbone) [7]	0.85	0.85	0.85	0.65	2,142,276	4,218,531
TDNet [19]	0.85	0.83	0.82	0.52	28,200,000	6,380,000,000
Video propagation and label relaxation[61]	0.89	0.88	0.88	0.72	137,100,096	91,055,000,000
FCN-8 + ConvLSTM [52]	0.86	0.86	0.85	0.61	134,629,100	269,821,618
U-Net + ConvLSTM	0.90	0.90	0.90	0.76	21,695,976	62,842,913
DeepLabV3+ + ConvLSTM	0.87	0.86	0.85	0.62	2,244,520	5,011,257
UVid-Net (U-Net encoder)	**0.91**	**0.91**	**0.91**	**0.79**	23,745,032	142,291,710
UVid-Net (ResNet-50 encoder)	0.90	0.89	0.89	0.72	44,740,420	133,871,366
UVid-Net(Transfer learning)	0.89	0.88	0.87	0.60	260	-

evaluate the performance of the proposed architecture, we compare it with the traditional U-Net architecture (with a single-encoder branch). Fig. 4 shows the comparison of the segmentation results obtained using a single-branch U-Net and two-branch UVid-Net. Since single-branch U-Net is an image semantic segmentation algorithm, it fails to capture the temporal information and, hence, produces temporally inconsistent labels. In contrast, the proposed architecture is able to capture the temporal dynamics between the two keyframes and produces more accurate results. For instance, the U-Net with single-branch encoder incorrectly classifies few pixels belonging to road/greenery class as construction (shown in yellow circles). However, the two-branch encoder-based proposed method correctly classifies these pixels as road/greenery, thus producing a temporally smoother segmentation result as shown in Fig. 4(d) and (e).

Tables I and II compare the performance of single-branch encoder U-Net with the proposed UVid-Net in terms of mIoU, precision, recall, and F1-score. It is observed that the per-class IoU of UVid-Net (U-Net encoder) for all the four classes are higher than the single-branch U-Net. Moreover, from Table I, it is observed that the proposed method has higher recall and precision scores than single-branch U-Net, which indicates that it has produced lower false positives and false negatives. The above results demonstrate the effectiveness of two-branch encoder module in acquiring temporal information and, thus, resulting in a more accurate segmentation as compared to the classical single-branch encoder U-Net. It may be noted that a

TABLE II
PER-CLASS IOU AND MIOU OF VARIOUS ALGORITHMS ON MANIPALUAVID DATASET

Method	IoU (Greenery)	IoU (Road)	IoU (Construction)	IoU (Water bodies)	mIoU
U-Net [42]	0.86	0.81	0.56	0.79	0.75
FCN-8 [30]	0.83	0.75	0.50	0.48	0.64
DeepLabV3+ [7]	0.79	0.75	0.59	0.47	0.65
TDNet [19]	0.78	0.72	0.51	0.08	0.52
Video propagation and label relaxation [61]	0.82	0.80	**0.67**	0.61	0.72
FCN-8 + ConvLSTM [52]	0.84	0.75	0.49	0.39	0.61
U-Net + ConvLSTM	0.87	0.82	0.56	0.82	0.76
DeepLabV3+ + ConvLSTM	0.81	0.76	0.57	0.28	0.62
UVid-Net (U-Net encoder)	0.87	**0.86**	0.60	**0.86**	**0.79**
UVid-Net (ResNet-50 encoder)	0.88	0.82	0.50	0.69	0.72
UVid-Net(Transfer learning)	**0.89**	0.80	0.54	0.2	0.60

The bold values represents the best performing algorithms.

single-branch U-Net with ResNet-50 encoder suffered from high variance (overfitting) even in the presence of regularization, with training and validation accuracies of 0.98 and 0.66, respectively.

Tables I and II also compare the performance of encoder architectures based on U-Net encoder and ResNet-50. It can be observed that the proposed encoder module based on U-Net encoder and ResNet-50 achieves a comparable performance (in terms of IoU, Table II) on greenery and road class while a slightly lower IoU is observed in construction and water bodies class for ResNet-50 based encoder. This decrease in IoU for water bodies and construction class for ResNet-50 based encoder is on expected lines due to the challenges encountered in learning the parameters of a deeper network (ResNet-50) with limited training images. The decrease in IoU for two classes results in a slightly lower mIoU for UVid-Net based on ResNet-50 encoder as compared to that of U-Net encoder. However, in spite of the decrease in IoU for two classes, the overall mIoU obtained using UVid-Net with ResNet-50 based encoder (0.72) is comparable with that of the current state-of-the-art method [61].

In addition to the qualitative and quantitative evaluation of the encoder, the softmax output of U-Net and UVid-Net (U-Net encoder) is also analyzed in Fig. 5. It can be observed that a high probability score is obtained for the pixels in their actual class in UVid-Net as compared to that of U-Net. The high probability score eliminates uncertainty and produces a more accurate segmentation. For example, a high probability score for greenery class is obtained for pixel belonging to trees using UVid-Net (Fig. 5). In addition, U-Net which lacks temporal information has produced higher construction class probability for pixel belonging to greenery at the boundaries (refer the 6×6 representative regions in Fig. 5). In contrast, the UVid-Net which utilizes features propagated from the previous frame has produced very low construction class probability for greenery pixels at the class boundaries.

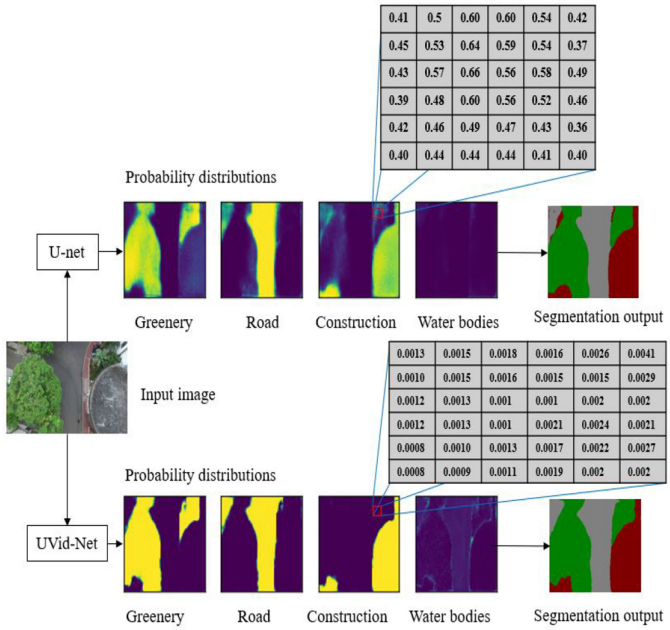


Fig. 5. Heat map of probability distributions produced by U-Net and UVid-Net algorithms. Row 1 shows the softmax output of U-Net, while row 2 shows the softmax output of UVid-Net. Also shown are the actual softmax output for a 6×6 region.

C. Evaluation of Decoder

The decoder of the proposed UVid-Net architecture consists of skip connections from the lower branch of the encoder to the corresponding decoder layers. Elementwise multiplication operation is utilized to combine the activations of the encoder and decoder layers. The experimental evaluation of the proposed feature-refiner module with the concatenation approach suggests a marginal increase in mIoU for UVid-Net with U-Net encoder (Tables III and IV). It can be observed that the per-class IoU is higher for road and water bodies for the multiplication operation as compared to concatenation. Further, the other two classes (greenery and construction) perform competitively in terms of per-class IoU. However, the qualitative evaluation shows that a more accurate segmentation is obtained using the proposed approach compared with the concatenation. Fig. 6 shows few images where finer segmentation boundaries are obtained using the UVid-Net (multiplication) with U-Net encoder as compared to UVid-Net (concatenation). It may be observed in Fig. 6 (first two rows) that the pixels from the road class have been misclassified as construction class using UVid-Net (concatenation), while a precise greenery–road boundary is obtained using UVid-Net (multiplication). The improvement obtained using the proposed feature-refiner module is more prominent for UVid-Net (ResNet encoder). An mIoU of 0.72 is obtained with UVid-Net (ResNet

TABLE III
COMPARING PERFORMANCE OF UVID-NET (CONCATENATION) WITH UVID-NET (MULTIPLICATION)

UVid-Net Variations	Precision	Recall	F1 -Score	mIoU	Learnable Parameters	FLOPs
U-Net encoder (Concatenation)	0.91	0.90	0.90	0.78	26,878,472	161,093,886
ResNet-50 encoder (Concatenation)	0.81	0.82	0..80	0.53	47,801,668	152,672,006
U-Net encoder (Multiplication)	**0.91**	**0.91**	**0.91**	**0.79**	23,745,032	142,291,710
ResNet-50 encoder (Multiplication)	0.90	0.89	0.89	0.72	44,740,420	133,871,366

The bold values represents the best performing algorithms.

TABLE IV
PER-CLASS IOU AND MIOU OF UVID-NET FOR COMPARING PERFORMANCE OF UVID-NET (CONCATENATION) WITH UVID-NET (MULTIPLICATION)

UVid-Net Variations	IoU (Greenery)	IoU (Road)	IoU (Constr-uction)	IoU (Water bodies)	mIoU
U-Net encoder (Concatenation)	0.88	0.83	0.60	0.84	0.78
ResNet-50 encoder (Concatenation)	0.84	0.72	0.17	0.40	0.53
U-Net encoder (Multiplication)	0.87	**0.86**	0.60	**0.86**	**0.79**
ResNet-50 encoder (Multiplication)	0.88	0.82	0.50	0.69	0.72

The bold values represents the best performing algorithms.

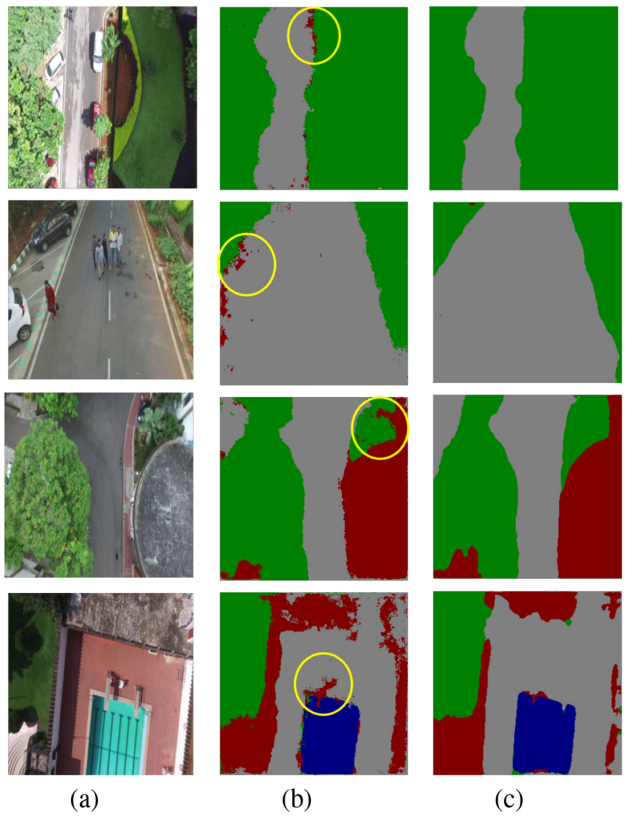


Fig. 6. Evaluating decoder: segmentation results obtained using UVid-Net (U-Net encoder, concatenation) and UVid-Net (U-Net encoder, multiplication). (a) Original image. (b) Results of concatenation of feature maps [UVid-Net (concatenation)]. (c) Results of elementwise multiplication [UVid-Net (multiplication)]. Note the improvement in the segmentation by utilizing elementwise multiplication (yellow circles). Here, green, gray, red, and blue colors represent the greenery, road, construction, and water bodies class, respectively.

encoder) with multiplication operation as compared to 0.53 with concatenation.

Moreover, the feature-refiner module reduces the number of FLOPs along with a number of parameters. It is observed that UVid-Net (multiplication) results in 142 291 716 FLOPs while UVid-Net (concatenation) results in 161 093 892 FLOPs for U-Net encoder (∼11% less FLOPs). These results show that an accurate segmentation is obtained using UVid-Net (multiplication) with much less computation overhead. Besides, the elementwise multiplication operation in UVid-Net also reduces the number of learnable parameters (23 745 032) in the network as compared to the concatenation in the UVid-Net (26 862 856). This result is significant since the proposed architecture produces higher mIoU (in the order of 0.79) with a reduced number of parameters. Indeed, the reduced complexity and the number of parameters of UVid-Net as compared to traditional concatenation operation makes it an ideal CNN architecture which can be used for UAV-based IoT applications.

D. Comparison With State of the Art

The proposed approach is compared with the existing state-of-the-art image semantic segmentation methods viz., U-Net [42], FCN8 [30], and DeepLabV3+ [7]. However, these methods do not incorporate temporal information and segment each keyframe independently. Therefore, the proposed method is also compared with the state-of-the-art approaches ([19] and [61]) on CityScape dataset that includes temporal information. The authors in [19] used multiple shallow CNNs to extract features from multiple frames. Subsequently, attention mechanism is utilized to combine the temporal features. In [61], the authors proposed to use video prediction model to propagate labels to the immediate neighboring frames for creating more image-label pairs [61]. Besides, the performance of UVid-Net is compared with the UAV *video* semantic segmentation approach proposed in [52]. This approach uses a convolution long short-term memory (ConvLSTM) module to capture the temporal dynamics of the video. It may be noted that the method proposed in [52] independently segments each frames using FCN8, and then the resulting frames are passed through ConvLSTM module as the postprocessing step. However, in addition to combining FCN8 + ConvLSTM, we also compare the performance by segmenting individual frames with U-Net/DeepLabV3+ and then postprocessing it with ConvLSTM module, resulting in two additional methods viz., UNet + ConvLSTM and DeepLabV3+ + ConvLSTM.

The proposed architecture is quantitatively compared with the above-mentioned existing approaches. Table I compares the performance metrics such as precision, recall, F1-score, and

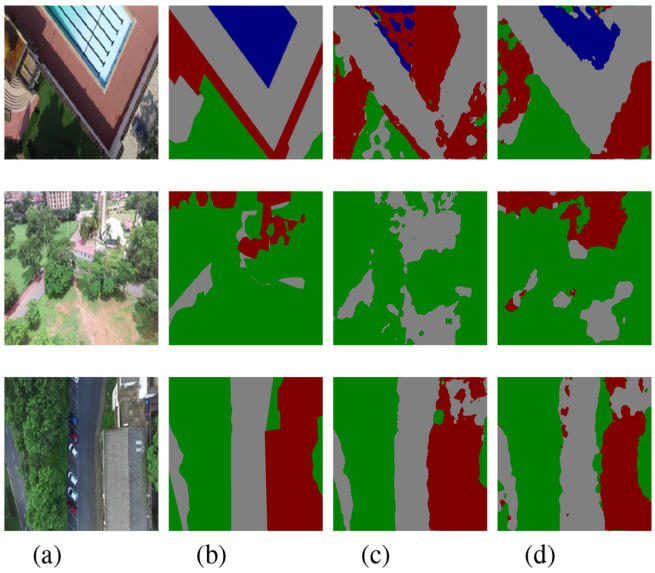


Fig. 7. Visual results comparison of UVid-Net and [61] on construction class. (a) Raw image. (b) Ground truth. (c) Results of [61]. (d) Results of UVid-Net.

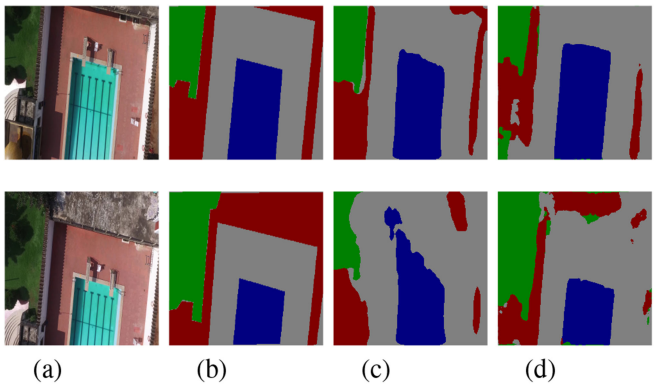


Fig. 8. Comparing performance of U-Net + ConvLSTM with UVid-Net (U-Net encoder). (a) and (b) Two consecutive key frames and its corresponding ground-truth. (c) and (d) Results of U-Net with ConvLSTM and UVid-Net. Here, green, gray, red, and blue colors represent the greenery, road, construction, and water bodies class, respectively.

mIoU, while Table II compares the per-class IoU and mIoU of the existing methods with the proposed method. As discussed earlier, the image semantic segmentation approaches (U-Net, FCN, and DeepLabV3+) segment each keyframe independently and fail to capture temporal cues. It can be observed that an mIoU of 0.79 is obtained by the proposed approach as compared to an mIoU of 0.75, 0.64, and 0.65 for U-Net, FCN8, and DeepLabV3+, respectively. The proposed approach outperforms the existing image segmentation approach. Besides, it can be observed from Fig. 9 that UVid-Net produces a more accurate segmentation map with smoother segmentation boundaries as compared with other approaches. The proposed UVid-Net incorporates temporal information by merging the features extracted from two different frames of a video and thereby outperforms the existing image semantic segmentation algorithms.

In addition to the image segmentation algorithms, the proposed approach is also compared with the video semantic segmentation algorithms viz., TDNet [19], video propagation/label relaxation [61], UNet-ConvLSTM, FCN8-ConvLSTM [52], and DeepLabV3+ - ConvLSTM. It can be seen (Table I) that the UVid-Net (U-Net encoder) achieves an mIoU of 0.79 and F1-score of 0.91 outperforming the other video segmentation approaches. Besides, UVid-Net (ResNet50-encoder) performs competitively and achieves an F1-score of 0.89 and an mIoU of 0.72. To study the performance of the proposed method for each class, the per-class IoU is computed as shown in Table II.

The water bodies class accounts for only 1.2% of the total pixels in the dataset. In spite of the limited annotation available, the proposed approach UVid-Net outperforms the existing methods and the current state of the art [61] by a significant margin (IoU of 0.86 for UVid-Net vs. 0.61 for [61]). Moreover, the construction class accounts for a slightly higher pixel count (5.5%) in the dataset. For the construction class, the proposed method outperforms the existing algorithms (except [61]) in terms of IoU. A slightly higher IoU is observed using [61] (0.67) as compared to UVid-Net (0.60) for the construction class. Fig. 7 compares the performance of [61] and UVid-Net for segmentation of construction class. More accurate segmentation is obtained for few frames using the current state-of-the-art [61] method for construction class as compared to the proposed method (Fig. 7, row 3). However, for other frames, a more accurate segmentation is obtained for the construction class using the proposed method (Fig. 7, rows 1 and 2). The proposed approach performs competitively (for construction class) with the current state-of-the-art [61] method with a significant reduction in the model parameters and without the need for an extra sequential model/optical flow. It may also be noted that [61] contains 137 M parameters, [19] contains 28 M parameters, while the proposed approach contains 23 M parameters. Besides, there is a reduction in the computational complexity (91 055 000 000 FLOPs for [61], 6 380 000 000 FLOPs for [19], while only 142 291 710 FLOPs for UVid-Net). Hence, the proposed method for video semantic segmentation is efficient in terms of computational complexity and is a viable solution for edge computing based applications such as scene parsing using UAV. Fig. 9 compares the segmentation results obtained using the proposed approach and the existing methods. It can be observed that the more accurate segmentation is obtained using the proposed method as compared to the existing methods. For instance, the proposed method is able to accurately identify construction, greenery, and water bodies especially in fifth and eight rows of Fig. 9.

The UNet-ConvLSTM performs competitively on ManipalUAVid dataset with an mIoU of 0.76. However, U-Net-ConvLSTM fails to capture the temporal dynamics as shown in Fig. 8. In comparison, UVid-Net (U-Net encoder) produces a more accurate segmentation, especially for the water body class.

In addition to the significant improvement in the performance, the UVid-Net (U-Net encoder) has a lower number of parameters as compared to the FCN-8, FCN-8 + ConvLSTM as shown in Table I. Further, UVid-Net (U-Net encoder) has a comparable number of parameters with other models with an exception of DeepLabV3+ which uses MobileNet-V2 backbone. The lower

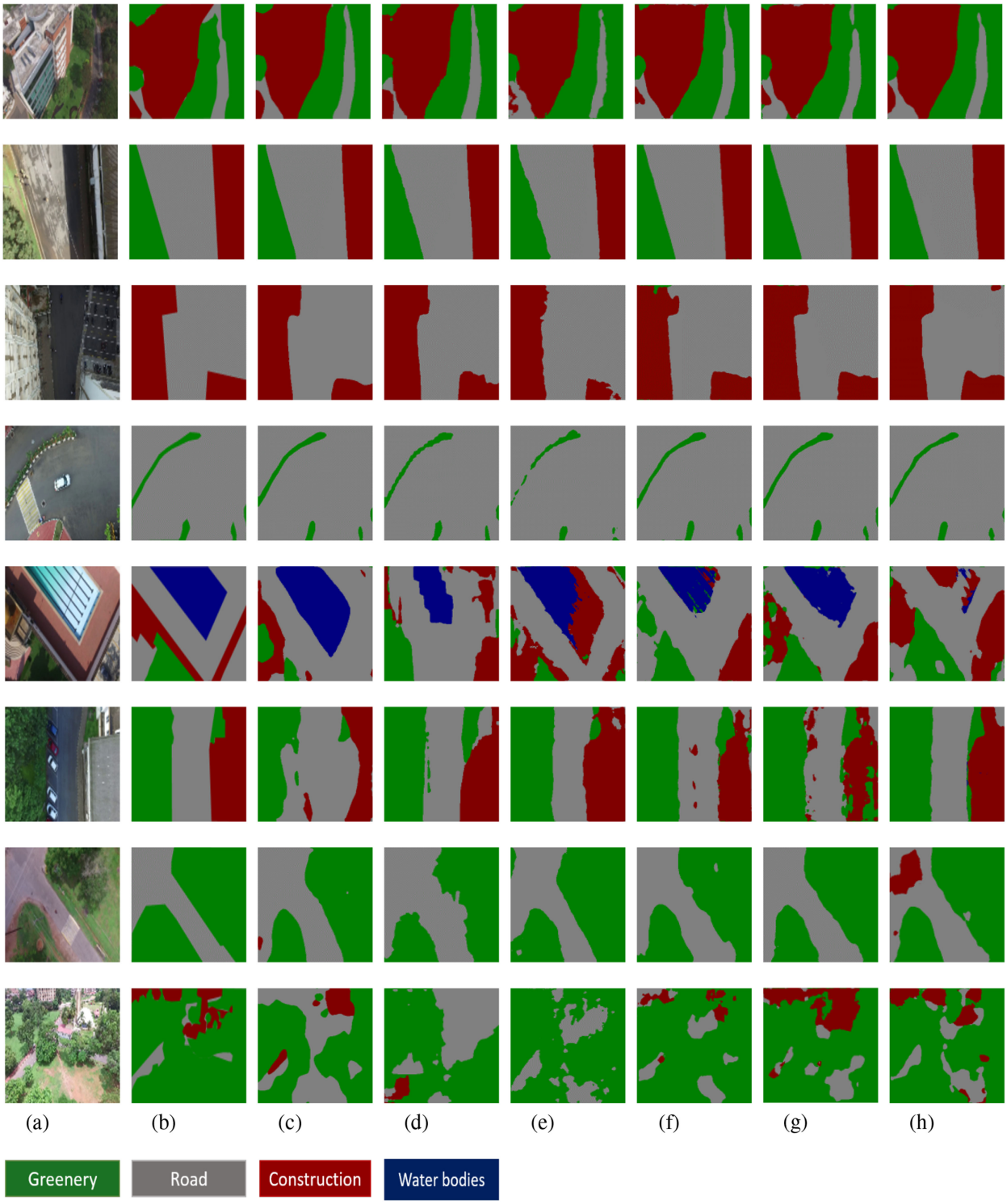


Fig. 9. UAV video semantic segmentation results on ManipalUAVid dataset [15]. (a) and (b) Keyframes from UAV video and its corresponding ground truth. (c) and (d) Results of ConvLSTM with U-Net and FCN8 backbone architectures, respectively. (e) Results of [61]. (f) Results of UVid-Net with ResNet50 encoder. (g) Results of UVid-Net with U-Net encoder. (h) Results of transfer learning.

parameters of UVid-Net reduces the dependency on the availability of huge training data.

E. Evaluation of Transfer Learning

The availability of manually annotated training dataset of sufficient size is a challenge in supervised deep learning based approach. A widely used approach in this scenario is to train the CNN network on a huge dataset and then transfer the weights learned for the task at hand [56].

In this work, the transfer learning approach has been studied on UVid-Net (U-Net encoder) for semantic segmentation of UAV aerial videos. The UAVid-Net (U-Net encoder) is initially

trained on Cityscape [11] dataset to predict eight categorical classes (flat, human, vehicle, construction, object, nature, sky, and void) by using Adam optimizer with a learning rate set to 0.0001. This dataset is selected due to its similarity in classes as compared to ManipalUAVid. Moreover, this dataset consists of 3000 training images which are greater than the ManipalUAVid dataset and helps in learning more generalized features. Subsequently, the last layer of the model is retrained (with other layers frozen) on the ManipalUAVid dataset to predict four classes (greenery, road, construction, and water bodies). The performance metrics of UVid-Net (U-Net encoder) by utilizing transfer learning is shown in Tables I and II. It is observed that the UVid-Net has performed competitively on greenery, road, and construction classes with a per-class IoU of 0.89, 0.80, and 0.54, respectively. However, a low per-class IoU is observed on water bodies class (0.20). This result was expected since the Cityscape dataset does not contain any images with water and has no definition for water bodies class. Fig. 9 shows the segmentation result of transfer learning on UVid-Net (U-Net encoder). It can be observed that the transfer learning approach offers competitive results as compared to existing approaches on greenery, road, and construction classes. Despite the limitation on unknown classes, pretrained UVid-Net (U-Net encoder) could be the preferred choice especially in the case of limited availability of training dataset for UAV aerial videos segmentation.

V. Conclusion

This article presents a new encoder–decoder based CNN architecture for semantic segmentation of UAV aerial videos. The proposed architecture utilizes a new encoder consisting of two parallel encoding branches with two consecutive keyframes of the video as the input to the network. By integrating the features extracted from the two encoding branches, the network can learn temporal information eliminating the need for an extra sequential module. Besides, it uses a feature-refiner module in the decoder path. This module produces smoother segmentation boundaries. The proposed architecture achieved an mIoU of 0.79 on ManipalUAVid dataset which outperforms the other state-of-the-art algorithms. This work also demonstrated that the proposed network UVid-Net trained on a larger semantic segmentation dataset for urban street scenes (Cityscape) can be utilized for UAV aerial videos segmentation. This transfer learning approach shows that competitive results are obtained on ManipalUAVid dataset by retraining only the last layer of UVid-Net trained on Cityscape dataset. These results hold significance as it reduces the dependency on the availability of manually annotated training dataset which is a time-consuming and laborious task. The improved efficiency of UVid-Net by incorporating temporal information, along with reduced dependency on the availability of training data, will provide better segmentation of aerial videos. The lightweight architecture of UVid-Net aids in reducing the computational complexity and number of trainable parameters, which makes it an ideal CNN architecture for UAV-based IoT applications. This improved segmentation can be utilized for monitoring of environmental changes, urban planning, disaster management, and other aerial surveillance tasks. In future, the developed system will be studied for real-time performance and be deployed in UAV drones for real-time scene analysis.

In general, commercially available UAVs are not flown with very high speed for applications such as scene analysis, surveillance, etc. The proposed model assumes a slow camera motion. In the presence of very large camera motion, there exist large scene variations between two consecutive frames. In these situations, estimation of temporal correspondence becomes mandatory for propagation of temporal information from frame to frame. However, the proposed work utilizing shot boundary detection and multibranch encoder has shown to be robust to small camera motion. Moreover, the proposed approach is a more suitable method for UAV-based IoT applications because of the reduction in the number of trainable parameters, computational complexity, and transferable features.

References

[1] A. Bansal, S. Ma, D. Ramanan, and Y. Sheikh, "Recycle-GAN: Unsupervised video retargeting," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 119–135.

[2] L. Bao, B. Wu, and W. Liu, "CNN in MRF: Video object segmentation via inference in a cnn-based higher-order spatio-temporal MRF," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 5977–5986.

[3] E. W. J. R. Bergsma, L. P. Almar, M. de Almeida, and M. Sall, "On the operational use of UAVs for video-derived bathymetry," *Coastal Eng.*, vol. 152, 2019, Art. no. 103527.

[4] D. Bulatov, P. Solbrig, H. Gross, P. Wernerus, E. Repasi, and C. Heipke, "Context-based urban terrain reconstruction from UAV-videos for geoinformation applications," *Int. Archives Photogramm. Remote Sens. Spatial Inf. Sci.*, vol. 38(1/C22), pp. 75–80, 2011.

[5] A. Y. C. Chen and J. J. Corso, "Temporally consistent multi-class video-object segmentation with the video graph-shifts algorithm," in *Proc. IEEE Workshop Appl. Comput. Vis.*, 2011, pp. 614–621.

[6] L.-C. Chen, 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 Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018.

[7] L.-C. Chen, Y. G. Zhu, F. P. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 801–818.

[8] H.-T. Cheng and N. Ahuja, "Exploiting nonlocal spatiotemporal structure for video segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 741–748.

[9] E. Corcoran, S. Denman, J. Hanger, B. Wilson, and G. Hamilton, "Automated detection of koalas using low-level aerial surveillance and machine learning," *Sci. Rep.*, vol. 9, no. 1, pp. 1–9, 2019.

[10] M. Cordts *et al.*, "The cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3213–3223.

[11] M. Cordts *et al.*, "The cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3213–3223.

[12] P. Fang, J. Lu, Y. Tian, and Z. Miao, "An improved object tracking method in UAV videos," *Procedia Eng.*, vol. 15, pp. 634–638, 2011.

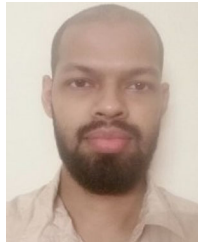
[13] M. Fayyaz, M. H. Saffar, M. Sabokrou, M. Fathy, F. Huang, and R. Klette, " STCFN: Spatio-temporal fully convolutional neural network for semantic segmentation of street scenes," in *Proc. Asian Conf. Comput. Vis.*, 2016, pp. 493–509.

[14] S Girisha, M. Pai, U. Verma, and R. Pai, "Semantic segmentation of UAV aerial videos using convolutional neural networks," in *Proc. 2nd IEEE Int. Conf. Artif. Intell. Knowl. Eng.*, 2019, pp. 21–27.

[15] S Girisha, M. M. Manohara, U. Pai Verma, and R. M. Pai, "Performance analysis of semantic segmentation algorithms for finely annotated new UAV aerial video dataset (ManipalUAVid)," *IEEE Access*, vol. 7, pp. 136239–136253, 2019.

[16] S. Girisha *et al.*, "Semantic segmentation of UAV videos based on temporal smoothness in conditional random fields," in *Proc. IEEE Int. Conf. Distrib. Comput., Elect. Circuits Robot.*, 2020, pp. 241–245.

[17] A. Gupta, E. Welburn, S. Watson, and H. Yin, "Post disaster mapping with semantic change detection in satellite imagery," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops,*2019, pp. 472–474.
[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
[19] P. Hu, F. Caba, O. Wang, Z. Lin, S. Sclaroff, and F. Perazzi, "Temporally distributed networks for fast video semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 8818–8827.
[20] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, "CCNet: Criss-cross attention for semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 603–612.
[21] S. Jain, X. Wang, and J. E Gonzalez, "Accel: A corrective fusion network for efficient semantic segmentation on video," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 8858–8867.
[22] X. Jin *et al.*, "Video scene parsing with predictive feature learning," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 5581–5589.
[23] A. Kae, B. Marlin, and E. Learned-Miller, "The shape-time random field for semantic video labeling," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 272–279.
[24] B. Kalantar, S. B. Mansor, A. Abdul Halin, H. Z. M. Shafri, and M. Zand, "Multiple moving object detection from UAV videos using trajectories of matched regional adjacency graphs," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 9, pp. 5198–5213, Sep. 2017.
[25] B. Kellenberger, D. Marcos, and D. Tuia, "When a few clicks make all the difference: Improving weakly-supervised wildlife detection in UAV images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2019, pp. 1414–1422.
[26] A. Kundu, V. Vineet, and V. Koltun, "Feature space optimization for semantic video segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3168–3175.
[27] J. Li, Y. Zhao, J. Fu, J. Wu, and J. Liu, "Attention-guided network for semantic video segmentation," *IEEE Access*, vol. 7, pp. 140680–140689, 2019.
[28] X. Lian, Y. Pang, J. Han, and J. Pan, "Cascaded hierarchical atrous spatial pyramid pooling module for semantic segmentation," *Pattern Recognit.*, vol. 110, 2021, Art. no. 107622.
[29] Y. Liu, D. Zhang, Q. Zhang, and J. Han, "Part-object relational visual saliency," *IEEE Trans. Pattern Anal. Mach. Intell.*, to be published, doi: 10.1109/TPAMI.2021.3053577.
[30] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3431–3440.
[31] C. Luo, W. Miao, H. Ullah, S. McClean, G. Parr, and G. Min, "Unmanned aerial vehicles for disaster management," in *Proc. Geological Disaster Monit. Based Sensor Netw.*, 2019, pp. 83–107.
[32] B. Mahasseni, S. Todorovic, and A. Fern, "Budget-aware deep semantic video segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2077–2086.
[33] O. Miksik, D. Munoz, J. Andrew Bagnell, and M. Hebert, "Efficient temporal consistency for streaming video scene analysis," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2013, pp. 133–139
[34] D. Ogawa *et al.*, "Surveillance of panicle positions by unmanned aerial vehicle to reveal morphological features of rice," *PLoS One*, vol. 14, no. 10, pp. 1–13, 2019.
[35] M. Pai, V. Mehrotra, S. Aiyar, U. Verma, and R. Pai, "Automatic segmentation of river and land in SAR images: A deep learning approach," in *Proc. IEEE 2nd Int. Conf. Artif. Intell. Knowl. Eng.*, 2019, pp. 15–20.
[36] M. Pai, V. Mehrotra, U. Verma, and R. M Pai, "Improved semantic segmentation of water bodies and land in SAR images using generative adversarial networks," *Int. J. Semantic Comput.*, vol. 14, no. 1, pp. 55–69, 2020.
[37] M. Paul, M. Danelljan, L. Van Gool, and R. Timofte, "Local memory attention for fast video semantic segmentation," 2021, *arXiv:2101.01715*.
[38] M. Paul, C. Mayer, L. Van Gool, and R. Timofte, "Efficient video semantic segmentation with labels propagation and refinement," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2020, pp. 2862–2871.
[39] X. Peng and C. Schmid, "Multi-region two-stream R-CNN for action detection," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 744–759.
[40] S. Qiao, R. Wang, S. Shan, and X. Chen, "Deep video code for efficient face video retrieval," in *Proc. Asian Conf. Comput. Vis.*, 2016, pp. 296–312.
[41] A. Rangnekar, N. Mokashi, E. Ientilucci, C. Kanan, and M. J. Hoffman, "Aerorit: A new scene for hyperspectral image analysis," vol. 58, no. 11, pp. 8116–8124, 2019.
[42] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Interv.*, 2015, pp. 234–241.
[43] J. Shotton, J. Winn, C. Rother, and A. Criminisi, "Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 1–15.
[44] A. Singh, H. Kalke, M. Loewen, and N. Ray, "River ice segmentation with deep learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 11, pp. 7570–7579, Nov. 2020.
[45] N. Souly, C. Spampinato, and M. Shah, "Semi supervised semantic segmentation using generative adversarial network," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 5688–5696.
[46] Y. Sun, Y. Hua, L. Mou, and X. X. Zhu, "Cg-net: Conditional GIS-aware network for individual building segmentation in VHR SAR images," *IEEE Trans. Geosci. Remote Sens.*, to be published, doi: 10.1109/TGRS.2020.3043089.
[47] U. Verma, F. Rossant, and I. Bloch, "Segmentation and size estimation of tomatoes from sequences of paired images," *Eurasip J. Image Video Process.*, vol. 2015, no. 1, pp. 1–23, Dec. 2015.
[48] B. Wang, L. Li, Y. Nakashima, R. Kawasaki, H. Nagahara, and Y. Yagi, "Noisy-LSTM: Improving temporal awareness for video semantic segmentation," *IEEE Access*, vol. 9, pp. 46810–46820, 2021.
[49] H. Wang, W. Wang, and J. Liu, "Temporal memory attention for video semantic segmentation," 2021, *arXiv:2102.08643*.
[50] J. Wang, Y. Zhang, J. Lu, and W. Xu, "A framework for moving target detection, recognition and tracking in UAV videos," in *Proc. Affect. Comput. Intell. Interact.*, 2012, pp. 69–76.
[51] L. Wang, W. Li, W. Li, and L. Van Gool, "Appearance-and-relation networks for video classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1430–1439.
[52] Y. Wang, Y. Lyu, Y. Cao, and M. Y. Yang, "Deep learning for semantic segmentation of UAV videos," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2019, pp. 2459–2462.
[53] Y. Wei, K. Zhang, and S. Ji, "Simultaneous road surface and centerline extraction from large-scale remote sensing images using CNN-based segmentation and tracing," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 12, pp. 8919–8931, May 2020.
[54] W. Wu, H. Li, X. Li, H. Guo, and L. Zhang, "Polsar image semantic segmentation based on deep transfer learning-realizing smooth classification with small training sets," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 6, pp. 977–981, Jun. 2019.
[55] Z. Xuerui and Y. Xia, "LSMVOS: Long-short-term similarity matching for video object," 2020, *arXiv:2009.00771*.
[56] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 3320–3328.
[57] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "Bisenet: Bilateral segmentation network for real-time semantic segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 325–341.
[58] H. Yu, J. Wang, Z. Huang, Y. Yang, and W. Xu, "Video paragraph captioning using hierarchical recurrent neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 4584–4593.
[59] F. G. Zanjani *et al.*, "Improving semantic video segmentation by dynamic scene integration," in *Proc. NCCV*, 2016, pp. 1–10.
[60] L. Zhou, C. Zhang, and M. Wu, "D-linknet: Linknet with pretrained encoder and dilated convolution for high resolution satellite imagery road extraction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2018, pp. 192–1924.
[61] Y. Zhu *et al.*, "Improving semantic segmentation via video propagation and label relaxation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 8848–8857.



Girisha S. (Member, IEEE) received the B.E degree from Srinivas School of Engineering, Visvesvaraya Technological University (VTU), Belgaum, India, in 2015, and the master's degree in computer science and engineering from N.M.A.M. Institute of Technology (NMAMIT), Nitte, India, in 2017. He is currently working toward the Ph.D. degree in computer vision and deep learning at Manipal Institute of Technology, Manipal, India.

His area of research interest is image segmentation, object detection, and deep learning for computer vision.

Ujjwal Verma (Senior Member, IEEE) received the M.S. (Research) degree in signal and image processing from IMT Atlantique, Nantes, France, in 2010, and the Ph.D. degree in image processing from Télécom ParisTech, University of Paris Saclay, Paris, France, in 2014.

He is currently an Associate Professor with the Department of Electronics and Communication Engineering, Manipal Institute of Technology. His research interests include variational methods in image segmentation, action recognition, and deep learning methods for scene understanding.

Dr. Verma is a recipient of "ISCA Young Scientist Award 2017–18" by Indian Science Congress Association (ISCA), a professional body under the Department of Science and Technology, Government of India. He is a recipient of "Young Professional Volunteer Award 2020" by IEEE Mangalore Sub-Section in recognition of his outstanding contribution to IEEE activities. He was the Joint Secretary, IEEE Mangalore Sub-Section for the year 2019.

Manohara Pai M. M. (Senior Member, IEEE) received the Ph.D. degree in computer science and engineering from University of Mysore, Mysore, India, in 2001.

He has been a Professor with the Department of Information and Communication Technology, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, India, for the past 29 years. He holds six patents to his credit and has authored or coauthored more than 100 papers in national and international journals/conference proceedings. He has authored or coauthored 2 books and guided 8 Ph.D. and 79 master theses. His areas of research interests include data analytics, cloud computing, IoT, computer networks, mobile computing, scalable video coding, and robot motion planning.

Dr. Pai is a life member of ISTE and life member of Systems Society of India. He is the Principal Investigator for multiple Industry/Govt. research projects. He is the Executive Committee Member of IEEE Mangalore Sub-Section and past Chair of IEEE Mangalore Sub-Section (2019).

Radhika M. Pai (Senior Member, IEEE) received the Ph.D. degree in data mining from National Institute of Technology Karnataka, Surathkal, India, in 2008.

She is a Professor with the Department of Information and Communication Technology, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, India. She has a teaching experience of more than 28 years. She has authored or coauthored 70 papers in national/international journals/conferences and has guided 5 Ph.D. and several master theses. Her areas of research interests include data mining, Big Data analytics, pattern recognition, sensor networks, and e-learning.

Dr. Pai is the recipient of National Doctoral fellowship from AICTE, Government of India. She is the Principal Investigator for a Government funded project and also Coinvestigator for some projects. She has been an Executive Committee Member of IEEE Mangalore Sub-Section for the last four years (2017–2020).