2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

Efficient Semantic Segmentation for Compressed Video

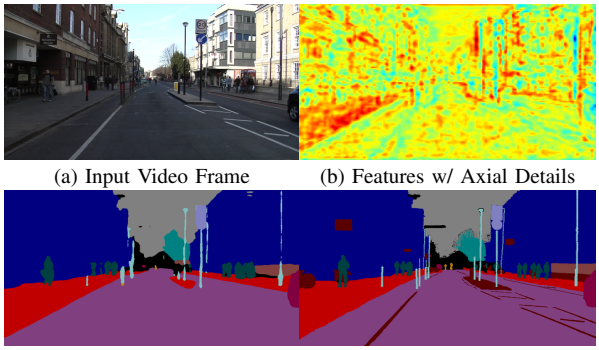
Jiaxin Cai[1], Qi Li[1], Yulin Shen[1], Jia Pan[2], Wenxi Liu[1]

***Abstract*— Robots, constrained by limited onboard computing resources, often encounter situations wherein high-resolution and high-bit-rate videos captured by their cameras necessitate compression before further analysis. In this paper, we propose a novel video semantic segmentation paradigm for compressed video. Specifically, our framework draws the inspiration from the principle of Wavelet Transform, and thus we design the network structure, WTDecomNet, approximating the decomposition of high-resolution image into its low-resolution counterpart and axial details. The aim is to well preserve the image content through decomposition and maintain model efficiency by obtaining semantics from low-resolution image. To facilitate this purpose, we propose an efficient axial subband approximation module for extracting axial details and a lightweight temporal alignment module for associating keyframes and non-keyframes of compressed video. Through comprehensive experiments, we show that our model can achieve the state-of-the-art performance on public benchmarks. Especially on CamVid, comparing to baseline, our proposed model reduces the computational overhead by $\sim 70\%$ while improving mIoU by $\sim 4\%$.**

I. Introduction

Semantic segmentation constitutes a foundational challenge within the realm of robotic vision systems [15], [17], [35]. Mobile robots or embedded AI systems, constrained by limited on-board computing resources, are often operated remotely or in a semi-autonomous manner. This implies that robot systems must be capable of communicating with base stations or cloud platforms [11] and transmitting at least parts of the acquired sensor data. A central issue in real-world environments is the insufficient network bandwidth, preventing the transmission of the entire high-resolution and high-bit-rate video stream to the remote system. For instance, a typical Kinect sensor generates an RGB-D video stream about 45 megabytes per second (MBps) [23], while the average bandwidth in a U.S. household is only 12.6 megabits per second (Mbps) [1]. This necessitates compressing the captured video stream before transmission, with many robotic applications [25]–[27], [31] commonly employing video compression standards such as VP8, H.264, and H.265.

The advantages of semantic segmentation predicated on compressed video are twofold, comparing to model optimization techniques [16], [21], [30]. First, it mitigates the difficulty of transmitting and storing large raw video data. Second, it does not demand sophisticated training procedure or model deployment like model distillation, network pruning, or quantization, to achieve efficient computation. Consequently, investigating this problem holds significant practical relevance within the field of robotics.



(a) Input Video Frame (b) Features w/ Axial Details

(c) Segmentation Result (d) Ground-truth

Fig. 1. In this work, our primary objective is to achieve semantic segmentation in the context of compressed video, utilizing a wavelet transform-guided image decomposition network. The model we propose is designed to capture axial details, thereby mitigating information loss resulting from image downsampling, ultimately leading to efficient and high-quality segmentation. As illustrated in the above figure, we present visualizations of the features containing axial details that substantially enhance the segmentation of slender objects, such as light poles, while concurrently mitigating compression artifacts. Note that, during the evaluation process, the regions highlighted in dark red are excluded from consideration.

Existing video semantic segmentation methods [7], [18], [34], [37] often incur significant overhead because of heavy temporal correlation modules (e.g., optical flow estimation), affecting their suitability for real-time applications. To improve efficiency, on the other hand, recent approaches [6], [9], [10], [13] have sought to enhance efficiency by leveraging the partition of keyframes and non-keyframes encoded in compressed video streams. These strategies predominantly prioritize computational efficiency at the expense of discarding non-keyframe information, with compensation for the lost data achieved exclusively through keyframes. But, the improper selection of keyframes easily results in degradation of performance or inference speed.

In contrast to prior methods, we propose a novel and extremely efficient video semantic segmentation paradigm specifically tailored for compressed video, without the need to discard the crucial information of non-keyframes while exploiting the temporal cues of compressed video. Specifically, in order to preserve the information within video frames by the largest extent, we are inspired by the principle of *Wavelet Transform* (WT) that losslessly decomposes an image into one approximation subband and three subbands of details with axial texture generics at a low-resolution scale. To this end, we design a network architecture, denoted Wavelet

This work was supported by the National Natural Science Foundation of China (Grant No. 6720110, U21A20471, U21A20472).

[1]J. Cai, Q. Li, Y. Shen, W. Liu are with College of Computer and Data Science, Fuzhou University, China. W. Liu is the corresponding author.

[2]J. Pan is with Department of Computer Science, the University of Hong Kong, HKSAR

Transform guided Decomposition Network (WTDecomNet). In concrete, our proposed model consists of two parallel network branches to align with the principles of WT. These branches comprise the following components: an approximation branch and an axial branch. The approximation branch is responsible for computing the semantic features of the low-resolution approximation subband. Meanwhile, the axial branch is equipped with Axial Subband Approximation (ASA) modules to facilitate the extraction of horizontal and vertical axial features from the original frame (see Fig. 1). As a result, our model effectively preserves semantic information while simultaneously achieving high inference speeds, thereby demonstrating its suitability for practical applications in the field of robotics.

To facilitate the real-time application on robotic systems, our proposed ASA modules and temporal alignment module are extremely lightweight. Since the ASA module models the horizontal and vertical details only, it costs linear complexity. In addition, the temporal alignment module exploits the motion vectors inherent in compressed videos at almost no additional cost to achieve temporal correlation.

In experiments, we demonstrate that our proposed model outperforms the state-of-the-art approaches in speed and accuracy on public benchmarks CamVid and Cityscapes. Ablation studies also demonstrate that our proposed new paradigm can better compensate for lost information, and the proposed module is both lightweight and effective. Our model can achieve 50 and 27 FPS for CamVid and Cityscapes on a single GPU, respectively.

In overall, our contributions are summarized as follows:

- We propose an effective and efficient video semantic segmentation framework for compressed video, which draws the inspiration from wavelet transform and thus designs the network structure to approximate image decomposition.
- To facilitate the effective and efficient segmentation model, we propose the axial subband approximation module that extracts horizontal and vertical details to compensate the information loss of image downscaling. Additionally, we incorporate an exceptionally lightweight temporal alignment module to establish temporal correlations across frames.
- Through comprehensive experiments on public benchmarks (i.e., CamVid and Cityscapes), it is demonstrated that our model achieves both state-of-the-art performance in terms of computational efficiency and segmentation accuracy. Comparing to the baseline approaches, our model reduces the computational overhead by $\sim 70\%$ while improving mIoU by $\sim 4\%$.

II. Related Works

We survey the related literature on video semantic segmentation and compressed-domain video analysis.

A. Video Semantic Segmentation

Existing methods dealing with video tasks tend to capitalize on temporal continuity in videos and thus to extract various kinds of temporal features, among which optical flow is commonly used [34], [37]. However, due to the extra time consumed by the flow estimation module, which is itself a bottleneck for real-time performance. Another group of methods [9], [22], [28] focus on utilizing temporal redundancy in videos. They propose to propagate the deep features extracted from keyframes to reduce the computation for non-keyframes. [28] proposes to directly reuse the segmentation results from keyframes, while Mahasseni et al. [22] interpolate segmentation results from the neighborhood. TD-Net [9] aggregates the features from different time stamps and replaces the deep model with several shallow models distributed across the timeline. Instead of processing the image frames as a whole, some methods [10], [14], [29] attempt to improve efficiency by reducing image resolution. However, all of the above methods rely only on using features from keyframes to enhance non-keyframes, yet their huge spatio-temporal inconsistency leads to suboptimal results. Unlike these previous methods, inspired by the wavelet transform, we achieve information complementation of image downsampling by modeling axial information.

B. Compressed-domain Video Analysis

Recently, compressed video have been recently utilized in vision tasks. The motion vectors and residual maps encoded in the compressed video are treated as additional modalities and directly fed into networks for video classification [3], vehicle counting [19], action recognition [33], etc. Jain et al. [12] design a bidirectional feature warping module harnessing the motion vectors of the preceding and succeeding keyframes. Feng et al. [6] replace a block of warped features with a local non-keyframe feature patch for further refinement. Hu et al. [10] adopt attention-based refinement scheme to enhance the features of non-keyframes. However, all of the above methods require high computation overhead to construct temporal correlations. In contrast, we construct temporal correlations with only lightweight operations to enhance intra- and inter-frame information.

III. Proposed Method

In this section, we first clarify the motivation and briefly introduce the network framework. Next, we elaborate the details of WTDecomNet. Last, we describe the Axial Subband Approximation and the Temporal Alignment module.

A. Motivation

Previous approaches have aimed to mitigate computational overhead by downsampling input video frames. However, these methods inevitably lead to performance degradation due to the substantial loss of visual information. Even when using specific frames as references to compensate for this information loss, suboptimal results may arise due to the considerable spatio-temporal inconsistency within the video. To address this challenge, wavelet transform provides insight into lossless image downsampling. Recall that wavelet transform can decompose a high-resolution image into low-frequency information and high-frequency details,

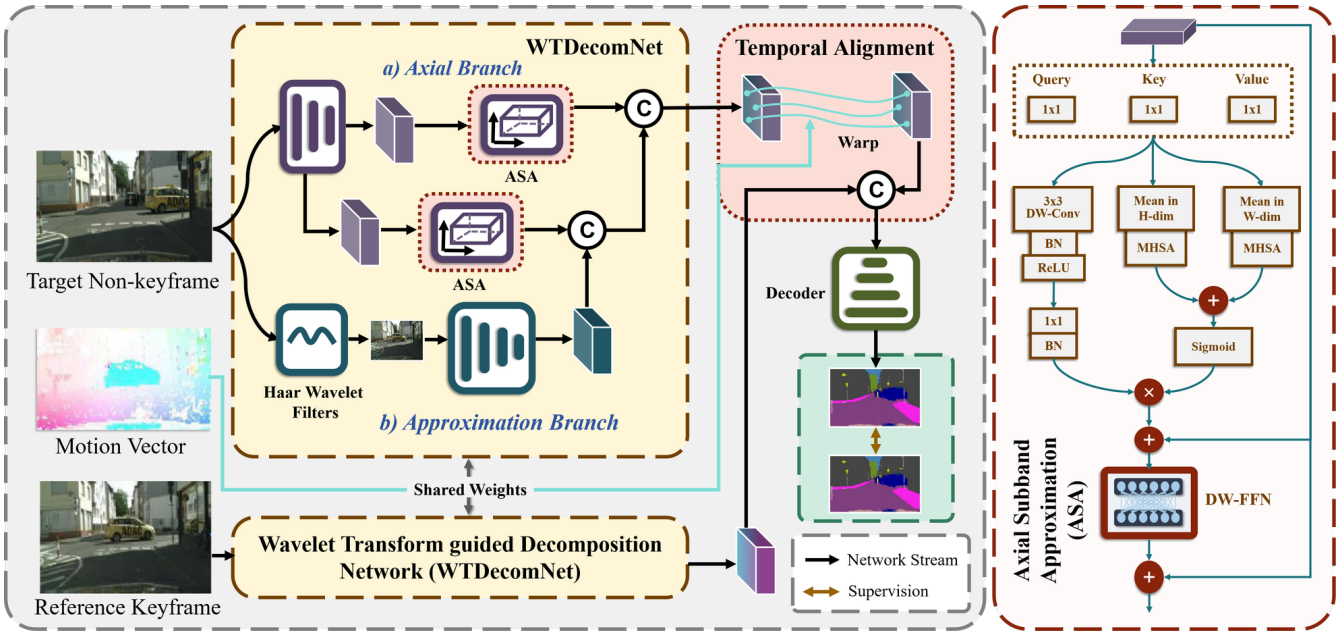


Fig. 2. Illustration of our framework. Each video frame is passed into a shared encoding network, WTDecomNet, which consists of two branches, approximation branch and axial branch. The approximation branch handles the wavelet transformed image downscaled by 4 times from the original frame to extract its coarse semantics. The axial branch captures the horizontal and vertical details via the ASA modules to compensate the semantics. Finally, the features of the target non-keyframe are fused with those of its reference keyframe, which has been warped for alignment purposes, facilitating the decoding of scene semantics.

using an approximation subband, i.e., a low-resolution image containing low-frequency content, and three detail subbands with axial texture generics in a lossless manner. Guided by the principles of wavelet transform, our model is designed to extract features from the downsampled image and axial details separately via our proposed WTDecomNet.

B. Framework Overview

Our framework is illustrated in Fig. 2. For each time step t ($t = \{1, \cdots, T\}$), the video frame $\mathcal{I}_t$ is fed into the shared network, i.e., WTDecomNet, to obtain the features X_t. Specifically, WTDecomNet consists of two branches, the approximation branch and the axial branch, which attempt to obtain the features of the approximation subband and axial subbands. Before sending the features X_t into the decoder, it will be merged with the warped features of the previous keyframe $\tilde{X}_{t-n}$, where n denotes the interval of GOP (i.e., Group of Pictures, which determines the interval for two adjacent keyframes). which is warped and aligned with the current frame according to the motion vector V_t inherent in the compressed video sequence.

C. WTDecomNet

In the WTDecomNet, a target non-keyframe $\mathcal{I}$ ($\mathcal{I} \in \mathbb{R}^{H \times W \times 3}$) of the given compressed video clip (we ignore the subscript t for brevity) is fed into the approximation branch and axial branch. Essentially, we model the low-resolution approximation subband of $\mathcal{I}$ and its detail subbands via these two branches, respectively. Hence, combining the approximation and detail subbands offers sufficient information for restoring the original video frame.

1) Approximation branch: To facilitate fast computation of semantic features, we preserve the low-frequency information of $\mathcal{I}$ and reduce the dimension of the original frame by 4×4 times. It can be accomplished by passing $\mathcal{I}$ through two subsequent Haar wavelet filters f_{LL} to obtain $\mathcal{I}_{LL}$ ($\mathcal{I}_{LL} \in \mathbb{R}^{\frac{H}{4} \times \frac{W}{4} \times 3}$), as expressed below.

$$\mathcal{I}_{LL} = \frac{1}{4} * (f_{LL} \otimes \frac{1}{4} * (f_{LL} \otimes \mathcal{I}) \downarrow 2) \downarrow 2, \tag{1}$$

$$f_{LL} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}, \tag{2}$$

where $\otimes$ denotes the convolution operator. Thus, $\mathcal{I}_{LL}$ is fed into the approximation branch. Then, employing an encoder with five residual convolution blocks with a pyramid pooling module, we can extract the features of the low-resolution approximation $\mathcal{I}_{LL}$.

2) Axial branch : In addition to the approximation of $\mathcal{I}$, the features of axial detail subbands via the axial branch should be extracted to compensate the lost information for downsampling the input image. Instead of directly feeding the detail subbands computed via Haar wavelet filters, we employ the original image at full resolution as the input of this branch to approximate high-frequency details. The reasons are twofold: 1) the original detail subbands contain little semantic information, which is detrimental to semantic segmentation; 2) there exists noises in the detail subbands that may be intrusive to semantic segmentation, e.g., overly dense intra-class texture.

To this end, we propose the Axial Subband Approximation (ASA) module to extract axial information from the original full-resolution video frame. Specifically, $\mathcal{I}$ is first passed

through a shallow encoder with four convolution blocks with downsampling layers, incurring a minimal additional computational cost. Next, as shown in Fig. 2, the extracted features will be delivered through the ASA modules to compute the features of axial details. Last, the axial features will join with the outcomes of the approximation branch for decoding.

3) Axial Subband Approximation: To thoroughly extract the axial details, the features from the 3-rd and 4-th stages of the encoder are both passed into ASA module. In essence, each ASA module models the global attention of the features along horizontal and vertical axes. As illustrated in the right part of Fig. 2, we show the detailed structure of the ASA module. In contrast to the typical multi-head self-attention module (MHSA), the ASA module comprises of two axial paths (i.e., horizontal and vertical paths) and a local path.

Formally, given the features $\mathbf{x} \in \mathbb{R}^{h\times w\times c}$ into the ASA module, we obtain the feature embeddings: $\mathcal{Q} = \Psi_q(\mathbf{x}), \mathcal{K} = \Psi_k(\mathbf{x}), \mathcal{V} = \Psi_v(\mathbf{x})$, where $\Psi_q(\cdot)$, $\Psi_k(\cdot)$ and $\Psi_v(\cdot)$ denote the tokenizers. For the axial paths, we squeeze the embeddings along the h- and w-dimension by computing the mean value, as below.

$$\begin{aligned}
&\mathcal{Q} \in \mathbb{R}^{h\times w\times C_{qk}} \Rightarrow \{\mathcal{Q}_h \in \mathbb{R}^{h\times C_{qk}}, \mathcal{Q}_v \in \mathbb{R}^{w\times C_{qk}}\},\\
&\mathcal{K} \in \mathbb{R}^{h\times w\times C_{qk}} \Rightarrow \{\mathcal{K}_h \in \mathbb{R}^{h\times C_{qk}}, \mathcal{K}_v \in \mathbb{R}^{w\times C_{qk}}\},\\
&\mathcal{V} \in \mathbb{R}^{h\times w\times C_{v}} \Rightarrow \{\mathcal{V}_h \in \mathbb{R}^{h\times C_{v}}, \mathcal{V}_v \in \mathbb{R}^{w\times C_{v}}\}. \qquad (3)
\end{aligned}$$

Hence, the subbands of the horizontal and vertical details can be approximated as:

$$\begin{aligned}
\mathcal{O} = &\text{MHSA}(\text{PE}_{\theta_1}(\mathcal{Q}_h), \text{PE}_{\theta_2}(\mathcal{K}_h), \mathcal{V}_h)\\
&+\text{MHSA}(\text{PE}_{\theta_3}(\mathcal{Q}_v), \text{PE}_{\theta_4}(\mathcal{K}_v), \mathcal{V}_v), \qquad (4)
\end{aligned}$$

where $\text{PE}_{\theta_i}(\cdot)$ denotes the learnable positional encoding. Since the ASA module handles the features for the horizontal and vertical axes only, its computational complexity is $\mathcal{O}((h^2+w^2)(C_{qk}+C_v)) \approx \mathcal{O}(hw)$ comparing to $\mathcal{O}(h^2w^2)$ of the original MHSA.

For the local path, in order to strengthen the spatial details, we stack the feature embeddings and employ 3×3 depth-wise convolution to extract the spatial information, and then a 1×1 convolution for fusing the channel-wise semantics.

D. Temporal Alignment

To obtain temporally consistent video semantic segmentation results, we leverage the features of compressed video, which innately partitions video frames into keyframes and non-keyframes and provides motion cues, i.e., motion vectors (MVs), at the block level. Since our model processes keyframes or non-keyframes without distinction, the spatial features of different frames can be easily fused using a very lightweight temporal alignment module ($\sim$ 0.06G) to construct temporal correlations efficiently.

Formally, for the extracted features of a non-keyframe as the target, X_{nk}, and its reference keyframe X_k, X_k is warped, denoted as X'_k, to align with X_{nk} subject to their motion vectors $\mathcal{M}_{nk}$. Hence, the fusion across frames can be achieved as:

$$X_{fusion} = \text{MLP}(\text{Concat}(X'_k, X_{nk})). \qquad (5)$$

Last, the fused features will be delivered into the decoder to produce the semantic segmentation results.

IV. Experimental Results

In this section, we conduct comprehensive experiments over public benchmarks. We thoroughly compare our model against the state-of-the-art methods to show the segmentation quality and computational cost, and perform ablation studies to demonstrate the effectiveness of our network structure.

A. Datasets and Implementation Details

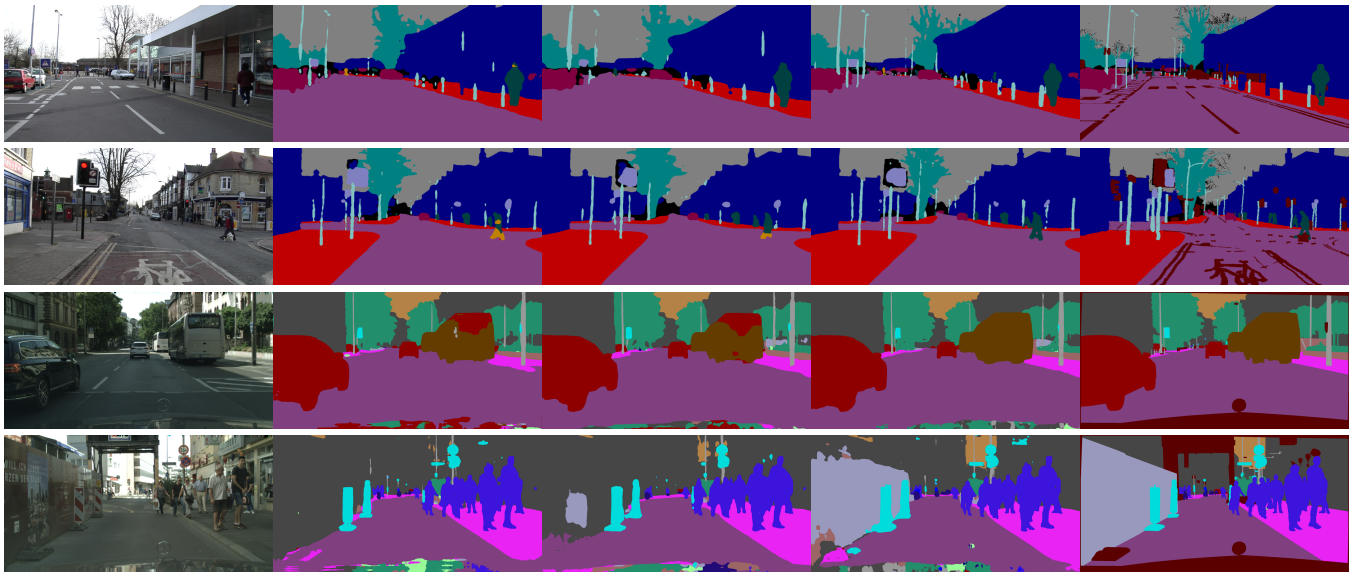
1) Datasets: We evaluate our proposed method on CamVid [2] and Cityscapes [4]. Following the experimental settings of [10], we compress the videos at reasonable bit-rates of 3Mbps for CamVid and 5Mbps for Cityscapes using the HEVC/H.265 standard. The length of GOP L defaults to 12.

2) Implementation Details: We implement our framework using PyTorch on a workstation with a single NVIDIA GeForce RTX 3090 GPU. During training, for CamVid we apply the Adam optimizer, and the initial learning rate is set to 0.001. For Cityscapes, we apply the SGD optimizer, and the initial learning rate is set to 0.01. The maximum epoch number is set to 100 and 200 for CamVid and Cityscapes, respectively. The mini-batch size is set to 8 and we use cosine annealing to decay the learning rate. For data augmentation, color jittering, horizontal flipping, random scaling, and random cropping are adopted. We employ the cross-entropy loss as the training objective. For CamVid and Cityscapes, our model yields 50 and 27 FPS, respectively.

3) Evaluation Metrics: All the comparison methods are evaluated on the compressed videos. Following [10], we evaluate our model with varying key distances d that measure the interval between the target frame p and the reference keyframe i. For $d = 0$, the frame p is a keyframe and it can be processed without temporal alignment. Otherwise, for $d \in (0, L-1]$, the features of the frame p will be aligned with those of i. The average of mIoU_d for different distances d is reported as the result. We measure GFLOPs using PyTorch-OpCounter. In addition to the accuracy and computational cost, we also follow [10], [24] to report the relative changes comparing to their single-frame backbone models. Specifically, $\widetilde{\Delta}$mIoU denotes the relative change of mIoU, and $\widetilde{\Delta}$GFLOPs denotes the relative change of GFLOPs.

B. Comparison with State-of-the-arts

We compare our model with the recent state-of-the-art methods for compressed video semantic segmentation over the benchmarks CamVid and Cityscapes. As shown in Table I, among all the comparison methods, our model achieves the optimal mIoU and GFLOPs. Considering the relative changes of mIoU and GFLOPs comparing the full model with its



(a) Input Image (b) PSPNet-18 [36] (c) AR-PSP18 [10] (d) Ours (e) Ground-truth

Fig. 3. We illustrate several examples from CamVid and Cityscapes, comparing our method against PSPNet-18 [36] and AR-PSP18 [10]. Note that, the particular regions with the color of dark red are not involved in the mIoU calculation.

TABLE I

COMPARISON RESULTS ON CAMVID TEST SET AND CITYSCAPES VALID SET. THE BEST RESULTS ARE HIGHLIGHTED IN BOLD AND THE SECOND BEST RESULTS ARE UNDERLINED.

	Method	mIoU(%)	GFLOPs	$\widetilde{\Delta}$mIoU	$\widetilde{\Delta}$GFLOPs
CamVid	Accel-DL18 [13]	66.15	397.70	**+13.8%**	+61.9%
	TD-PSP18 [9]	70.13	363.70	+1.0%	+17.7%
	BlockCopy [29]	66.75	107.52	-5.2%	-45.7%
	TapLab-BL2 [6]	67.57	117.73	-3.1%	-50.2%
	Jain et al. [12]	67.61	146.97	-4.3%	-53.8%
	AR-PSP18 [10]	70.82	133.09	+2.0%	-57.0%
	PSPNet-18 [36]	69.36	309.02	-	-
	Ours	**73.18**	**85.50**	+5.5%	**-72.3%**
Cityscapes	Accel-DL18 [13]	68.25	1011.75	**+18.4%**	+96.0%
	TD-PSP18 [9]	**70.11**	673.06	+1.6%	+20.0%
	BlockCopy [29]	67.69	294.20	-6.7%	-41.2%
	TapLab-BL2 [6]	68.90	237.29	-4.1%	-50.6%
	Jain et al. [12]	68.57	342.67	-5.1%	-52.5%
	AR-PSP18 [10]	69.45	234.91	+0.7%	-58.1%
	PSPNet-18 [36]	69.00	560.97	-	-
	Ours	69.88	**212.70**	+1.3%	**-62.0%**

single-frame baseline, our model not only improves the model performance but also reduces its computation cost. In concrete, our model performs better than the baseline model PSPNet-18 [36] while saving about 70% of computational cost. On the contrary, it is observed that some comparison methods (e.g., TDNet [9] and Accel [13]) usually improve accuracy ($\widetilde{\Delta}$mIoU $>$ 0), but also cost more computation resources ($\widetilde{\Delta}$GFLOPs $>$ 0), due to the introduction of the heavy temporal correlation module. In addition, other methods including BlockCopy [29], TapLab [6], and Jain et al. [12], sacrifice the model performance ($\widetilde{\Delta}$mIOU $<$ 0) in order to reduce the amount of computation burden ($\widetilde{\Delta}$GFLOPs $<$ 0). Note that, the recent method AR-PSP18 [10] also gains better mIoU while increasing efficiency. But, as the model heavily relies on keyframes to supplement the lost information, the spatio-temporal inconsistency leads to

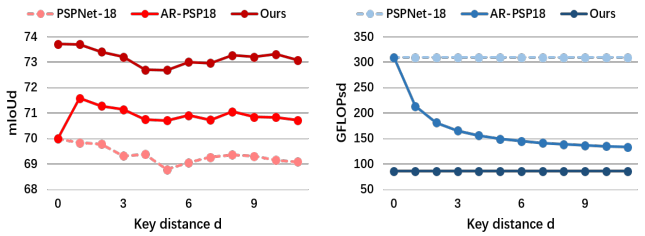
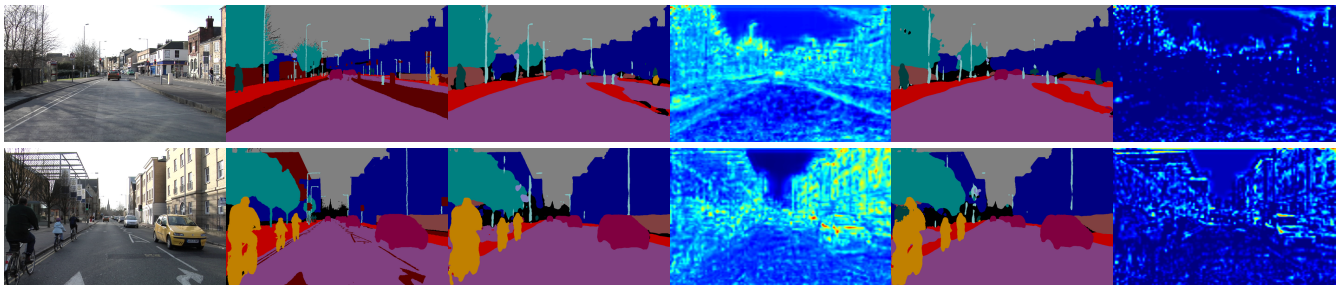


Fig. 4. We illustrate the mIoU$_d$ and GFLOPs$_d$ of different key distances d between keyframes for PSPNet-18, AR-PSP18, and our approach.

sub-optimal results. In contrast, our model compensates the lost information in the downsampled images via axial details, and efficiently constructs temporal correlations, leading to state-of-the-art results in accuracy and computation cost.

As shown in Fig. 3, we illustrate several examples of comparison on CamVid and Cityscapes between our approach and the state-of-the-art method, AR-PSP18 [10]. As observed, the competing methods tend to produce noisy predictions in detail regions (e.g. thin poles on the first row, small pedestrians on the second row) or misidentify large regions (e.g. they falsely recognize the wall as building on the fourth row) for compressed frames. Our proposed paradigm leverages low-resolution approximation as global context to recognize large regions, avoiding over-segmentation. In addition, we utilize the ASA module to mine the axial cues and accomplish both inter- and intra-frame information supplementation.

Temporal Gap To investigate the influence of the key distance d to the keyframe, we plot the mIoU and GFLOPs results for our model, PSPNet-18 [36], and AR-PSP18 [10] at different key distances d in Fig. 4. When $d > 0$, the accuracy decreases, since the compression artifacts in non-keyframes are more severe than those in keyframes. In contrast, our model benefits from the ASA and TA modules for mining details and complementing spatio-temporal information to maintain high accuracy for varying d. On the other hand,



(a) Input Image (b) Ground-truth (c) w/ ASA (d) Feat. w/ ASA (e) w/o ASA (f) Feat. w/o ASA

Fig. 5. We show examples of the results and features from the models with and without ASA.

it can be observed that the FLOPs of AR-PSP18 are on par with PSPNet-18 as the interval between keyframes is small, due to the difference in large computational expense required to process keyframes and non-keyframes. On the contrary, due to our extremely lightweight temporal alignment module, the latency of processing keyframes and non-keyframes is almost the same, which ensures that our model is almost unaffected by varying key distances while maintaining a high computational efficiency.

C. Ablation Study

We perform ablation studies to delve into the effectiveness of each component in our framework. All the ablation experiments are conducted on the test set of CamVid.

1) Model Structure: We investigate the efficacy of our proposed framework structure, ASA module, and temporal alignment module via ablation studies (see Table II). The first row represents the model containing the approximation branch only which is based on PSPNet-18. It can be seen that feeding the full resolution image to the baseline achieves good performance, but it results in large computational overhead. Downsampling the image to a quarter of the original dimension by Haar wavelet filters (HW-filters) greatly reduces the FLOPs but results in a severe performance degradation around 9%. To compensate the loss, we introduce the axial branch. For reference, we find that the model with a single axial branch containing a shallow encoder achieves comparable mIoU to the baseline while significantly decreasing the computation cost. In addition, incorporating the proposed ASA modules to mine the axial details, the model can further be strengthened by around 3% in terms of mIoU. The lightweight temporal alignment (TA) module is used to construct the temporal correlation by utilizing the motion vector from the compressed video, which adds up only about 0.06G of GFLOPs but gains obvious performance improvement.

2) Axial Subband Approximation Module: Inspired by the wavelet transform, we model the axial information complementary to the approximation branch in the axial branch via the ASA module. We also try different attention schemes in the axial branch as alternatives: 1) channel-spatial attention [32]; 2) window self-attention [20]; 3) self-attention [5]; 4) axis-attention [8]. As shown in Table III, our proposed ASA module achieves the best results. According to the principle of WT, the features obtained by the axial branch should be supplemented to those of the approximation branch

TABLE II

EFFECTIVENESS OF COMPONENTS IN OUR PROPOSED WTDECOMNET.

Approx.	HW-filters	Axial	ASA	TA	mIoU(%)	GFLOPs
✓					69.36	309.02
✓	✓				60.60	20.16
		✓			66.96	31.26
✓	✓	✓			69.05	77.08
✓	✓	✓		✓	70.00	77.14
✓	✓	✓	✓		71.81	85.44
✓	✓	✓	✓	✓	73.18	85.50

TABLE III

ABLATION STUDY ON THE ATTENTION SCHEME OF THE ASA MODULE.

Method	mIoU(%)	GFLOPs
w/o ASA	70.00	77.14
Channel-spatial attention [32]	70.83	80.46
Window self-attention [20]	71.10	83.49
Self-attention [5]	72.02	143.17
Axis-attention [8]	72.85	87.69
Ours	73.18	85.50

to accomplish lossless decomposition. Thus, other attention schemes may capture the information irrelevant or redundant towards the low-resolution approximation, leading to suboptimal results. The axis-attention [8] captures the correlation between each pixel and each axis, so its performance also exceeds that of the other approaches. But, it is less efficient and shortage of local contexts. By contrast, the local details and global axial cues are captured in our proposed ASA module via local path and axial paths, respectively. Our ASA module yields more accurate results and less computation cost. We also showcase an example with and without ASA module in Fig. 5. It is observed that our model can better depict spatial details of categories such as "Pole" and "Bicyclist" due to the supplement of axial information from ASA module.

V. CONCLUSION

In this paper, we propose a novel video semantic segmentation paradigm for compressed video. Specifically, our framework draws inspiration from the principle of Wavelet Transform, and thus we design WTDecomNet that approximates the decomposition of the high-resolution image into its low-resolution counterpart and axial details. Through comprehensive experiments, we show that our model can achieve state-of-the-art performance in terms of compressed video segmentation performance and efficiency.

REFERENCES

[1] D. Belson, J. Thompson, J. Sun, R. Möller, M. Sintorn, and G. Huston, "The state of the internet," *Akamai, Cambridge, MA, Tech. Rep*, 2015.

[2] G. J. Brostow, J. Fauqueur, and R. Cipolla, "Semantic object classes in video: A high-definition ground truth database," *Pattern Recognition Letters*, vol. 30, no. 2, pp. 88–97, 2009.

[3] A. Chadha, A. Abbas, and Y. Andreopoulos, "Compressed-domain video classification with deep neural networks:"there's way too much information to decode the matrix"," in *2017 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2017, pp. 1832–1836.

[4] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3213–3223.

[5] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[6] J. Feng, S. Li, X. Li, F. Wu, Q. Tian, M.-H. Yang, and H. Ling, "Taplab: A fast framework for semantic video segmentation tapping into compressed-domain knowledge," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 3, pp. 1591–1603, 2020.

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

[8] J. Ho, N. Kalchbrenner, D. Weissenborn, and T. Salimans, "Axial attention in multidimensional transformers," *arXiv preprint arXiv:1912.12180*, 2019.

[9] P. Hu, F. Caba, O. Wang, Z. Lin, S. Sclaroff, and F. Perazzi, "Temporally distributed networks for fast video semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8818–8827.

[10] Y. Hu, Y. He, Y. Li, J. Li, Y. Han, J. Wen, and Y.-J. Liu, "Efficient semantic segmentation by altering resolutions for compressed videos," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 22 627–22 637.

[11] J. Ichnowski, K. Chen, K. Dharmarajan, S. Adebola, M. Danielczuk, V. Mayoral-Vilches, N. Jha, H. Zhan, E. Llontop, D. Xu, *et al.*, "Fogros2: An adaptive platform for cloud and fog robotics using ros 2," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 5493–5500.

[12] S. Jain and J. E. Gonzalez, "Fast semantic segmentation on video using block motion-based feature interpolation," in *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, 2018, pp. 0–0.

[13] S. Jain, X. Wang, and J. E. Gonzalez, "Accel: A corrective fusion network for efficient semantic segmentation on video," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8866–8875.

[14] B. Kim, J. Yim, and J. Kim, "Highway driving dataset for semantic video segmentation," *arXiv preprint arXiv:2011.00674*, 2020.

[15] S. Kumaar, Y. Lyu, F. Nex, and M. Y. Yang, "Cabinet: Efficient context aggregation network for low-latency semantic segmentation," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 13 517–13 524.

[16] F. Li, C. Fu, F. Lin, Y. Li, and P. Lu, "Training-set distillation for real-time uav object tracking," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 9715–9721.

[17] S. Li, Q. Yan, C. Liu, M. Liu, and Q. Chen, "Holoseg: An efficient holographic segmentation network for real-time scene parsing," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 2395–2402.

[18] Y. Li, J. Shi, and D. Lin, "Low-latency video semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5997–6005.

[19] X. Liu, Z. Wang, J. Feng, and H. Xi, "Highway vehicle counting in compressed domain," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3016–3024.

[20] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 10 012–10 022.

[21] M. Mahajan, T. Bhattacharjee, A. Krishnan, P. Shukla, and G. C. Nandi, "Robotic grasp detection by learning representation in a vector quantized manifold," in *2020 International conference on signal processing and communications (SPCOM)*. IEEE, 2020, pp. 1–5.

[22] B. Mahasseni, S. Todorovic, and A. Fern, "Budget-aware deep semantic video segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1029–1038.

[23] F. Nenci, L. Spinello, and C. Stachniss, "Effective compression of range data streams for remote robot operations using h. 264, ieee," in *RSJ International Conference on Intelligent Robots and Systems, Sep*, 2014, pp. 14–18.

[24] M. Paul, M. Danelljan, L. Van Gool, and R. Timofte, "Local memory attention for fast video semantic segmentation," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 1102–1109.

[25] H. Qiao, T. Liu, and Y. Lu, "Indoor localization system for mobile robot based on visible light communication," in *2023 IEEE 18th Conference on Industrial Electronics and Applications (ICIEA)*. IEEE, 2023, pp. 269–274.

[26] A. Sau, A. Bhattacharyya, M. Ganguly, and S. K. Mahato, "An edge-inclusive webrtc-based framework to enable embodied visual analytics in telerobot," in *2023 15th International Conference on COMmunication Systems & NETworkS (COMSNETS)*. IEEE, 2023, pp. 228–230.

[27] M. Schwarz, C. Lenz, R. Memmesheimer, B. Pätzold, A. Rochow, M. Schreiber, and S. Behnke, "Robust immersive telepresence and mobile telemanipulation: Nimbro wins ana avatar xprize finals," *arXiv preprint arXiv:2303.03297*, 2023.

[28] E. Shelhamer, K. Rakelly, J. Hoffman, and T. Darrell, "Clockwork convnets for video semantic segmentation," in *Computer Vision–ECCV 2016 Workshops: Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part III 14*. Springer, 2016, pp. 852–868.

[29] T. Verelst and T. Tuytelaars, "Blockcopy: High-resolution video processing with block-sparse feature propagation and online policies," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 5158–5167.

[30] H. Wang, P. Cai, Y. Sun, L. Wang, and M. Liu, "Learning interpretable end-to-end vision-based motion planning for autonomous driving with optical flow distillation," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 13 731–13 737.

[31] Y. Wang, W. Wang, D. Liu, X. Jin, J. Jiang, and K. Chen, "Enabling edge-cloud video analytics for robotics applications," *IEEE Transactions on Cloud Computing*, 2022.

[32] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 3–19.

[33] C.-Y. Wu, M. Zaheer, H. Hu, R. Manmatha, A. J. Smola, and P. Krähenbühl, "Compressed video action recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 6026–6035.

[34] Y.-S. Xu, T.-J. Fu, H.-K. Yang, and C.-Y. Lee, "Dynamic video segmentation network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 6556–6565.

[35] Q. Yan, S. Li, C. Liu, M. Liu, and Q. Chen, "Fdlnet: Boosting real-time semantic segmentation by image-size convolution via frequency domain learning," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 8155–8162.

[36] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2881–2890.

[37] X. Zhu, Y. Xiong, J. Dai, L. Yuan, and Y. Wei, "Deep feature flow for video recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2349–2358.