Video-kMaX: A Simple Unified Approach for Online and Near-Online Video Panoptic Segmentation

Inkyu Shin[1,2†] Dahun Kim[2] Qihang Yu[2†] Jun Xie[2] Hong-Seok Kim[2] Bradley Green[2]
In So Kweon[1] Kuk-Jin Yoon[1] Liang-Chieh Chen[2†]
[1]KAIST [2]Google Research

Abstract

Video Panoptic Segmentation (VPS) aims to achieve comprehensive pixel-level scene understanding by segmenting all pixels and associating objects in a video. Current solutions can be categorized into online and near-online approaches. Evolving over the time, each category has its own specialized designs, making it nontrivial to adapt models between different categories. To alleviate the discrepancy, in this work, we propose a unified approach for online and near-online VPS. The meta architecture of the proposed Video-kMaX consists of two components: within-clip segmenter (for clip-level segmentation) and cross-clip associater (for association beyond clips). We propose clip-kMaX (clip k-means mask transformer) and LA-MB (location-aware memory buffer) to instantiate the segmenter and associater, respectively. Our general formulation includes the online scenario as a special case by adopting clip length of one. Without bells and whistles, Video-kMaX sets a new state-of-the-art on KITTI-STEP and VIPSeg for video panoptic segmentation Code and models are available at this link.

1. Introduction

Video Panoptic Segmentation (VPS) [19] aims at a holistic video understanding of the scene by unifying two critical and challenging tasks: semantically segmenting images and associating segmented regions across all frames in a video [42]. It can benefit various real-world applications, such as autonomous driving, robot visual control, and video editing.

With the rapid growth of interest, there have been several methods [19, 20, 24, 32, 43, 57] proposed for VPS. They can be categorized into online and near-online approaches, which process the video either frame-by-frame or clip-by-clip (a clip contains only a few consecutive video frames). The online approaches, such as VPSNet [19] and Video K-Net [24], segment each frame sequentially via the modern image-level segmenter [13, 47, 55], and build an additional association branch trained to enforce consistent predictions between frames [24, 49]. On the other hand, the near-online approaches, such as ViP-DeepLab [32] and TubeFormer [20], extend the modern image-level segmenter [9, 38] to process a clip by designing extra modules (*e.g.*, next-frame instance segmentation [32] or latent memory [20]). The clip-level predictions are then stitched [32] to form the final video segmentation results. The modules designed for online or near-online approaches are not only evolving over time, but also becoming very distinct for each scenario. Consequently, it is infeasible to easily adapt on-

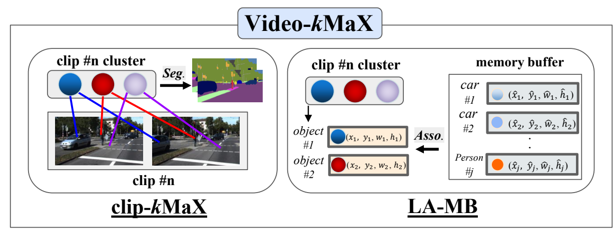


Figure 1. The meta architecture of Video-kMaX consists of clip-kMaX for clip-level segmentation and LA-MB for object association. The former groups pixels of the same object within-clip and the latter leverages appearance and location features (encoded by box coordinates) for long-term association across-clips.

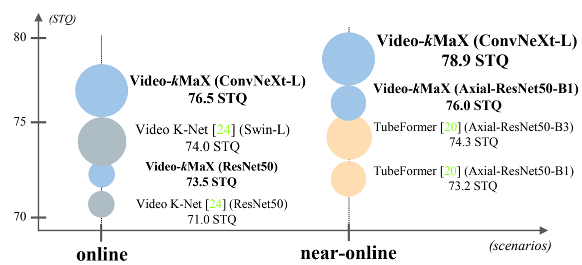


Figure 2. Video-kMaX is a unified approach for online and near-online video panoptic segmentation, showing state-of-the-art performance in both scenarios (on KITTI-STEP *val* set). The size of the circle reflects the model parameters.

[†] Work done while at Google Research.

line models to near-online (and vice versa). Particularly, the current online methods [19, 24] lack a proper clip-level segmenter, while the modern near-online methods [20, 32] fail to associate objects in an online manner, suffering from the absence of overlapping frames. The need for this scenario-specific design results in inefficiencies, as it requires the utilization of different frameworks for each setting. A natural question thus emerges: *Is it possible to develop a unified framework for online and near-online VPS without any scenario-specific design?*

To answer the question, we carefully design Video-kMaX, a simple yet effective approach for both online and near-online VPS. As drawn in Fig. 1, the meta architecture of Video-kMaX contains two components: within-clip segmenter and cross-clip associater, where the former component performs clip-level segmentation and the later one associates detected objects across clips. The proposed Video-kMaX is an instantiation of the pipeline by adopting clip-kMaX (clip k-means mask transformer) for the within-clip segmenter, and LA-MB (Location-Aware Memory Buffer) for the cross-clip associater.

The proposed clip-kMaX extends the image-level k-means mask transformer [52] to the clip-level *without* adding any extra modules or loss functions. Motivated by the k-means clustering perspective [51], we consider object queries as cluster centers, where each query is responsible for grouping pixels of the same object *within* a clip together. Specifically, each object query, when multiplied with the clip features [35, 38, 40], is learned to yield a *tube* prediction (*i.e.*, masks of the same object in a clip) [20]. This learning can be achieved via a surprisingly simple modification in the k-means cross-attention module [52] by concatenating the clip-level pixel features along the spatial dimension. As a result, clip-kMaX can be applied to both near-online and online settings without additional complexities. We also empirically show that k-means cross-attention is an effective mechanism for handling the extremely long sequence of spatially and temporally flattened clip features.

The proposed LA-MB is motivated by the drawbacks of existing methods through the careful systematical studies. We observe that the modern VPS methods [19, 32] could not handle the more challenging setting of *long-term* object tracking, since they either associate objects in the neighboring frames [19] or stitch overlapping frame predictions [32], making it hard to track objects beyond the short clip length. One promising solution is to exploit a memory buffer to propagate the tracking information across all video clips, which has been proven successful in the recent works [16, 45, 49, 53]. However, surprisingly, we observe that naïvely adopting the memory buffer to VPS leads to minor improvements or even worse performance. The setback enforces us to further look into its root case. We discover that the appearance feature alone [19, 45] is not sufficient for long-term association in VPS, when the target object is occluded for a long time; additionally, the memory buffer approach accumulates many detected objects, resulting in a huge matching space (between stored and newly detected objects) and hindering the matching accuracy. To resolve the issues, we develop LA-MB (Location-Aware Memory Buffer), which effectively incorporates location information to the memory module by two means. First, when comparing the similarity between the stored objects in memory and the detected objects in the current frame, we consider not only their appearance features (encoded by object queries), but also their location features (encoded by normalized bounding box coordinates). Specifically, if the object of interest is not detected in the current frame but it is stored in the memory (*e.g.*, due to occlusion), we will "predict" its current location by assuming the object is moving at a constant velocity. Second, we propose a *hierarchical* matching scheme to effectively reduce the matching space. We initially exploit the matching results from the Video Stitching [32] strategy, which associates objects based on their mask IoU in the overlapping frame between clips, effective for short-term association. We then associate the objects stored in memory with the currently detected but *unmatched* objects, aiming for long-term association. Thanks to our careful design, the LA-MB improves the long-term association quality both in near-online and online scenarios with low sensitivity to the hyper-parameter values.

In summary, we introduce Video-kMaX, a simple and unified method for online and near-online VPS. Our approach, consisting of two seamless modules: clip-kMaX and LA-MB, achieves significant performance improvements on two long sequence VPS datasets: KITTI-STEP [42] and VIPSeg [30]. In particular, as shown in Fig. 2, our best Video-kMaX outperforms the previous state-of-the-art online model (Video K-Net [24]) and near-online model (TubeFormer [20]) by **+2.5%** STQ and **+4.6%** STQ, respectively, on KITTI-STEP val set.

2. Related Work

Video Panoptic Segmentation (VPS) Video Panoptic Segmentation [19] aims to unify video semantic [12, 18, 31, 58] and video instance [1, 17, 37, 49] segmentation. Numerous efforts have been to transform image panoptic segmentation models [8, 21, 22, 25, 26, 38, 39, 48, 50] to the video domain. Among them, the online method VPSNet [19] adopts task-specific prediction heads from instance segmentation [13], semantic segmentation [11], and tracking [49], and jointly trains them to obtain panoptic video results. Similarly, the near-online method ViP-DeepLab [32] adds a next-frame instance segmentation head on top of Panoptic-DeepLab [9] that provides generic image panoptic segmentation with dual-ASPP [5] semantic segmentation module and dual-decoder [6] based instance segmentation. More

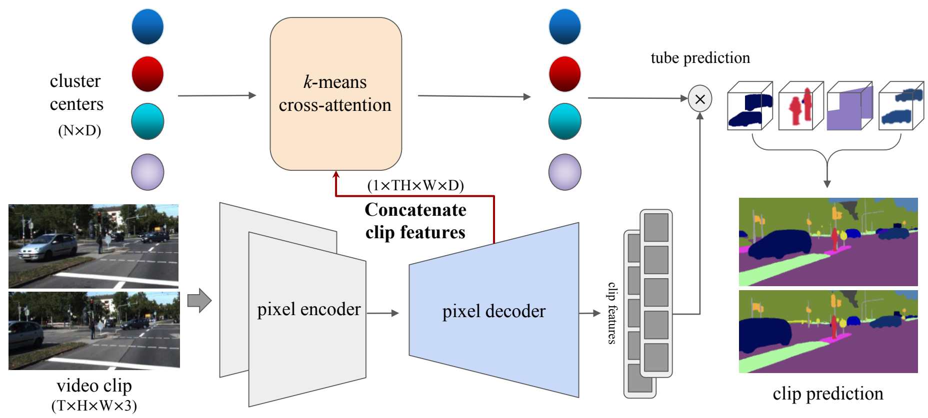


Figure 3. **Illustration of clip-kMaX.** The proposed clip-kMaX seamlessly converts the image-level segmentation model kMaX-DeepLab to clip-level *without* adding extra module. Motivated by the k-means perspective, clip-kMaX considers one object query as one cluster center, which learns to group together pixels of the same object *within* a clip. Specifically, each object query, when multiplied with the clip features, is learned to generate a *tube* prediction (*i.e.*, masks of the same object in the clip). This learning can be accomplished by a surprisingly simple change in the k-means cross-attention module by concatenating the clip-level pixel features along the spatial dimension (*i.e.*, treating the clip-level pixel features with shape $T \times H \times W \times D$ as one large image-level pixel feature with shape $1 \times TH \times W \times D$), where the input video clip contains T frames with height H and width W. D is the channel dimension of pixel features, and N is number of cluster centers.

recent works [20, 24, 57] identifies the limitations of previous methods that require multiple separate networks and complex post-processing (*e.g.*, NMS, fusion for tracking). To address the issues, they design a transformer architecture [3] for end-to-end video panoptic segmentation. However, all these methods have two fundamental issues. First, they require specific designs for either online or near-online scenario, *e.g.*, another association module [19, 24], temporal consistency loss [20, 24], or clip-segmentation module [20, 32]. Second, the models could only deal with short-term association (*i.e.*, either neighboring frames or a clip). In this regard, we propose a simple unified online and near-online video panoptic segmentation model for long-term association without adding extra scenario-specific designs.

Memory Module for Long-Term Tracking Object queries from the Transformer decoder [3] have been used to track objects in multi-object tracking [2,29,34,53,56], video instance segmentation [7,15–17,44–46], and video panoptic segmentation [20, 24, 57]. Some of them exploit queries for the short-term association [29, 34], while the others for the long-term association by additionally exploiting the memory buffer [45, 53]. Particularly, MOTR [53] proposes a set of track queries to model the tracked objects in the entire video. MeMOT [2] develops a spatio-temporal memory that stores a long range states of all tracked objects. MaskTrack R-CNN [49] employs a memory module to track detected objects. To make the association robust to challenging scenarios, such as heavy occlusion, IDOL [45] proposes a temporally weighted softmax score for object matching. Along the same direction, we specialize the memory buffer approach for both online and near-online video panoptic segmentation models, and additionally develop an efficient hierarchical matching scheme.

3. Method

The meta architecture of Video-kMaX contains two components: clip-kMaX (clip k-means mask transformer) for within-clip segmentation (Sec. 3.1) and LA-MB (location-aware memory buffer) for cross-clip association (Sec. 3.2). We detail them below, starting from the near-online framework. Our general formulation includes the online scenario by using clip length one (Sec. 3.3).

3.1. Within-Clip Segmenter: clip-kMaX

We first present the general formulation for image and video panoptic segmentation, before introducing our within-clip segmenter clip-kMaX, which performs clip-level segmentation with a short length T (*e.g.*, $T = 2$).

General Formulation for Image and Video Recently, image panoptic segmentation has been reformulated as a simple set prediction powered by Transformer [36]. From the pioneering works (*e.g.*, DETR [3] and MaX-DeepLab [38]) to the recent state-of-the-art methods (*e.g.*, kMaX-DeepLab [52]), panoptic predictions are designed to match the ground truth masks by segmenting image $I \in \mathbb{R}^{H \times W \times 3}$ into a fixed-size set of N class-labeled masks:

$$\{\hat{y}_i\}_{i=1}^{N} = \{(\hat{m}_i, \hat{p}_i(c))\}_{i=1}^{N}, \tag{1}$$

where $\hat{m}_i \in [0,1]^{H \times W}$ and $\hat{p}_i(c)$ denote predicted mask and semantic class probability for the corresponding mask, respectively. Motivated by this, TubeFormer [20] extends this formulation into set prediction of class-labeled *tubes*: $\{\hat{y}_i\}_{i=1}^N = \{(\hat{m}_i, \hat{p}_i(c))\}_{i=1}^N$, where $\hat{m}_i \in [0,1]^{T\times H\times W}$. In this setting, N object queries attend to the $T \times H \times W$ clip features, and predict N tubes. The prediction generalizes well for different values of T, since the positional embedding is only performed in the frame level, providing a useful structural prior that the same object in neighboring frames (assuming slow motion) will still be assigned by the same object query. Given the generalizability, we are able to absorb the T-axis into the H-axis before feeding the clip features to transformer decoder. Specifically, we propose to relax Eq. (1) into a more general form: $\{\hat{y}_i\}_{i=1}^N = \{(\hat{m}_i, \hat{p}_i(c))\}_{i=1}^N$, where $\hat{m}_i \in [0,1]^{S\times W}$, S=TH, and $T \geq 1$ (*i.e.*, S can change according to the different number of frames T). By doing so, it allows us to easily extend an image panoptic segmentation model to the video domain (clip-level), as detailed below.

clip-kMax The state-of-the-art image segmentation model kMaX-DeepLab [52] replaces the cross-attention in a typical transformer decoder [36] with k-means cross-attention by taking a cluster-wise argmax as below:

$$\hat{C} = C + \arg\max_N (Q^c \times (K^p)^T) \times V^p, \tag{2}$$

where $C \in \mathbb{R}^{N\times D}$ refers to N object queries with D channels. We use superscripts p and c to indicate the feature projected from the pixel features and object queries, respectively. $Q^c \in \mathbb{R}^{N\times D}, K^p \in \mathbb{R}^{HW\times D}, V^p \in \mathbb{R}^{HW\times D}$ stand for the linearly projected features for query, key, and value, respectively. In this k-means perspective, one object query is regarded as a cluster center, which learns to group pixels of the same object together. Given our previous general formulation, we can seamlessly extend kMaX-DeepLab to video clip, forming our clip-kMaX, by *simply* reshaping the key and value into $K^p \in \mathbb{R}^{SW\times D}$ and $V^p \in \mathbb{R}^{SW\times D}$ ($S = TH$ and T$\geq$1). The reshaping merges the T-frame feature to a *single*-frame feature with large height TH (*i.e.*, reshape $T \times H \times W$ to $1 \times TH \times W$), which then becomes compatible with the image model kMaX-DeepLab. This is equivalent to performing the k-means clustering for a video clip with length T, where one query is now learning to group pixels of the same object *in the clip* together. We illustrate clip-kMaX in Fig. 3. Note that kMaX-DeepLab then becomes a special case of clip-kMaX with $T = 1$.

Discussion The design of clip-kMaX may look simple on the surface. However, we made strenuous efforts in enhancing the conventional cross-attention module for clip-level mask predictions during its development. When dealing with the extremely large sequence length of spatially and temporally flattened clip features in a video clip, the standard cross-attention module is susceptible to learning. This phenomenon was evident in the poor performance of the original cross-attention, motivating the prior art TubeFormer [20] to further employ an additional latent memory module. To address this challenge, we propose using the k-means cross-attention [52] approach, which is capable of handling flattened clip features of any size by performing a cluster-wise argmax on N cluster centers.

Video Stitching (VS) In practice, given the limited memory, we are only able to perform clip-level inference (*i.e.*, segmenting a short clip with length T). To obtain the video-level segmentation, some heuristic designs are required. One popular approach is Video Stitching (VS) [20, 32], which propagates object identities between clips by matching the mask IoU scores in the *overlapping* frames. In our framework, we adopt the same video stitching strategy for our near-online Video-kMaX, but additionally explore memory buffer for long-term association.

3.2. Cross-Clip Associater: LA-MB

Our LA-MB basically consists of two phases: Encoding Phase to store the previous object features, and Decoding Phase to associate current objects with the objects stored in the memory buffer. We detail the process below.

Encoding Phase The memory buffer is initially empty, when a new testing video comes. It encodes features from all detected objects, while processing frames sequentially. Regarding the object features to be stored, we exploit the appearance and location properties of each object.

For the appearance feature of object i observed at frame t, we utilize the query embedding $q_i^t \in \mathbb{R}^D$ (*i.e.*, object queries from the mask transformer decoder [52]). The memory buffer encodes appearance feature $\hat{q}_i^t$ as follows:

$$\hat{q}_i^t = \begin{cases} (1-\lambda)\hat{q}_i^{t-1} + \lambda q_i^t, & \text{if } i \textbf{ both} \text{ in memory and frame } t, \\ q_i^t, & \text{else if } i \textbf{ only} \text{ in frame } t, \\ \hat{q}_i^{t-1}, & \text{else if } i \textbf{ only} \text{ in memory}, \end{cases} \tag{3}$$

where λ is the moving average weight between the stored appearance feature in memory $\hat{q}_i^{t-1}$ and current appearance feature q_i^t. We set λ to 0.8 as the default value.

Unlike other works [19, 45], we additionally exploit the location feature of object i observed at frame t, using its normalized bounding box (inferred from the predicted mask): b_i^t=$[x_i^{tl}/w, y_i^{tl}/h, x_i^{br}/w, y_i^{br}/h] \in \mathbb{R}^4$, where (x^{tl}, y^{tl}) and (x^{br}, y^{br}) are the x-y coordinates of top-left and bottom-right corners, and w and h denote the bounding box width and height, respectively. The memory buffer then encodes the location features as follows:

$$\hat{b}_i^t = \begin{cases} b_i^t, & \text{if } i \text{ in frame } t, \\ \hat{b}_i^{t-1} + (\hat{b}_i^{t-1} - \hat{b}_i^{t-2}), & \text{else if } i \textbf{ only} \text{ in memory}. \end{cases} \tag{4}$$

As shown in the equations, if an object is detected, the memory buffer will use its latest normalized bounding box information. If the object i is not detected but it is stored in the

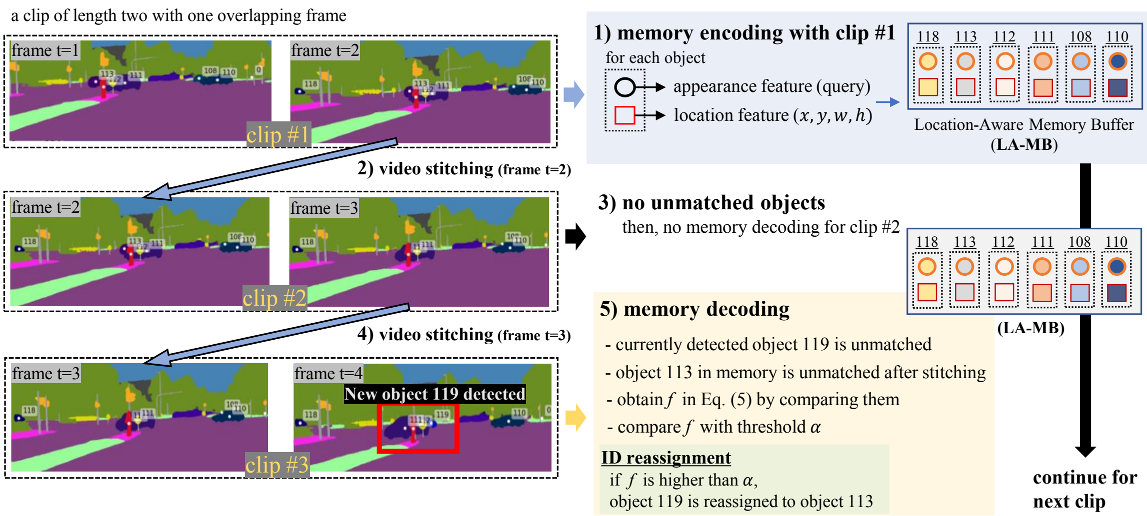


Figure 4. **Step-by-step overview of Location-Aware Memory Buffer (LA-MB).** The LA-MB approach consists of two phases: Encoding and Decoding. In the Encoding Phase, appearance and location features of detected objects are stored in the memory buffer. In the Decoding Phase, LA-MB performs hierarchical matching, beginning with video stitching for short-term association in overlapping frames between clips, followed by long-term association between unmatched objects in the current clip (*i.e.*, object 119 in figure) and objects stored in memory (*i.e.*, object 113 in figure).

memory (*e.g.*, due to occlusion), we will "predict" its current location by assuming the object's moving velocity is constant, *i.e.*, its location is shifted by $(\hat{b}_i^{t-1} - \hat{b}_i^{t-2})$ from its previous stored location $\hat{b}_i^{t-1}$.

Finally, the memory buffer stores both the appearance and location features $(\hat{q}_i, \hat{b}_i)$ for all M objects detected until the current frame. In practice, we adopt the memory refreshing strategy [45], where the old objects, whose last appeared frame is τ frame behind the current frame, are removed from the memory buffer. We empirically choose the optimal value for τ in our experiments.

Decoding Phase To specialize the memory buffer approach in our framework, we initially conduct the Video Stitching (VS) for *short-term* association between clips. Afterwards, we associate the objects stored in memory with the currently detected but *unmatched* objects, aiming for *long-term* association. This *hierarchical* matching mechanism forms our proposed **L**ocation-**A**ware **M**emory **B**uffer (LA-MB). Specifically, we compute the similarity function $f(i, j)$ between the currently *unmatched* object i (after VS) and the *encoded* object j in the memory as follows:

$$f(i, j) = e^{-\|b_i - \hat{b}_j\|^2/T} \cdot cos(q_i, \hat{q}_j). \tag{5}$$

We compute the negative L_2 distance between two normalized bounding boxes, weighted by a temperature T for scaling the values between location and appearance similarity. The appearance similarity is measured by the *cosine* distance. Then, we obtain a similarity matrix $\mathbf{S} \in \mathbb{R}^{M \times N}$ between M objects in memory and N detected objects in the current frame. To find the association, we perform Hungarian matching [23] on $\mathbf{S}$. Additionally, to filter out false associations, we only consider the matching with similarity value larger than a confidence threshold α. The unmatched objects in current frame are considered as new objects. The proposed LA-MB is illustrated in Fig. 4.

Discussion Our proposed LA-MB is partially inspired by the success of IDOL [45] in video instance segmentation, and memory buffer has been proven effective in several recent works [2,49,53]. However, there are two critical issues, if one naïvely applies their memory buffer approach to our framework (we name this method as naïve Memory Buffer (naïve-MB) for our baseline). First, the location feature is not exploited, but only the appearance feature. In a dynamic scene, object location plays an important role. The appearance feature becomes less reliable if the target object has been occluded for a long time. Second, the memory size M keeps growing as time goes by. Even though this issue is slightly alleviated by the memory refreshing strategy, it still results in a large matching space between the stored M objects in the memory and the currently detected N objects, which subsequently makes the one-to-one matching harder. To overcome the issues, our LA-MB proposes a novel formulation to incorporate the location features (Eq. (4) and Eq. (5)), and additionally augments the matching accuracy by performing the Video Stitching (VS) in the beginning of decoding phase, which effectively further reduces the matching space and improves the matching accuracy.

method	backbone	SQ	AQ	STQ
online methods				
Video K-Net [24]	ResNet50	71.0	70.0	71.0
Video K-Net [24]	Swin-L	75.0	73.0	74.0
Video-kMaX (online)	ResNet50	75.0	72.0	73.5
Video-kMaX (online)	ConvNeXt-L	**77.2**	**75.7**	**76.5**
near-online methods				
Motion-DeepLab [42]	ResNet50	67.0	51.0	58.0
TubeFormer [20]	Axial-ResNet50-B1	78.1	68.6	73.2
TubeFormer [20]	Axial-ResNet50-B3	78.3	70.0	74.3
Video-kMaX (near-online)	ResNet50	74.2	74.2	74.2
Video-kMaX (near-online)	Axial-ResNet50-B1	75.8	76.3	76.0
Video-kMaX (near-online)	ConvNeXt-L	**79.0**	**78.8**	**78.9**

(a) KITTI-STEP *val* set.

method	SQ	AQ	STQ
Motion-DeepLab [42]	59.8	45.6	52.2
Video K-Net [24]	65.0	60.0	63.0
TubeFormer [20]	**70.3**	60.6	65.3
UW_IPL/ETRI_AIRL [54]†	64.0	**71.3**	67.6
Video-kMaX (near-online)	69.8	67.2	**68.5**

(b) KITTI-STEP *test* set. †: ICCV 2021 challenge winning entry.

Table 1. [VPS] KITTI-STEP *val* and *test* set results.

method	backbone	SQ	AQ	STQ	VPQ
online methods					
VPSNet-FuseTrack [19]	ResNet50	-	-	20.8	17.0
VPSNet-SiamTrack [43]	ResNet50	-	-	21.1	17.2
Video K-Net [24] (arXiv version)	ResNet50	-	-	33.1	26.1
Video K-Net [24] (arXiv version)	Swin-base	-	-	46.3	39.8
Video-kMaX (online)	ResNet50	46.3	32.4	38.7	36.8
Video-kMaX (online)	ConvNeXt-L	**60.7**	**40.2**	**49.4**	**49.4**
near-online methods					
ViP-DeepLab [32]	ResNet50	-	-	22.0	16.0
Clip-PanoFCN [30]	ResNet50	-	-	31.5	22.9
TubeFormer [20] (arXiv version)	Axial-ResNet50-B1	50.3	31.6	39.8	29.2
TubeFormer [20] (arXiv version)	Axial-ResNet50-B3	53.0	32.5	41.5	31.2
Video-kMaX (near-online)	ResNet50	45.1	35.3	39.9	38.2
Video-kMaX (near-online)	Axial-ResNet50-B1	55.6	37.8	45.8	46.7
Video-kMaX (near-online)	ConvNeXt-L	**61.4**	**43.5**	**51.7**	**51.9**

(a) VIPSeg *val* set.

method	STQ	VPQ
Clip-PanoFCN [30]	25.0	22.9
TubeFormer [20] (arXiv version)	38.6	26.8
Video-kMaX (near-online)	**47.1**	**45.0**

(b) VIPSeg *test* set in the latest test server.

Table 2. [VPS] VIPSeg *val* and *test* set results.

3.3. Online Video Panoptic Segmentation

The meta architecture of Video-kMaX enables a general framework for both online and near-online VPS. When processing a clip of length one, our model performs online VPS. Specifically, the model is trained frame-by-frame and evaluated sequentially with the assistance of clip-kMaX's general formulation. Unlike the near-online setting, we skip the Video Stitching, which becomes infeasible in the online framework. Afterwards, we apply our LA-MB without any further modification.

4. Experimental Results

We conduct experiments on two *long sequences* Video Panoptic Segmentation datasets: KITTI-STEP [42] and VIPSeg [30].

4.1. Datasets

KITTI-STEP [42] is a Video Panoptic Segmentation (VPS) dataset that contains long video sequences with average track length 51 frames and maximum 643 frames, presenting a challenging scenario for long-term association. It contains 19 semantic classes, similar to Cityscapes [10], while only two classes ('pedestrians' and 'cars') come with tracking IDs. We adopt the Segmentation and Tracking Quality (STQ) as a metric for evaluation.

VIPSeg [30] is a new large-scale Video Panoptic Segmentation (VPS) benchmark providing in-the-wild real-world scenarios with 232 scenes and 124 classes, Among them, 58 classes are annotated with tracking IDs. The average sequence length is 24 frames per video. We adopt the STQ and VPQ [19] metric for evaluation.

4.2. Implementation Details

The proposed Video-kMaX is a unified approach for online and near-online VPS. For the near-online setting, we employ a clip length of two with one overlapping frame between clips. For the online setting, we set clip length to one and remove the video stitching strategy in the pipeline.

We employ two common backbones for both online and near-online settings: ResNet50 [14] and ConvNeXt-L [28]. We also experiment with Axial-ResNet50-B1 [39] backbone to fairly compare with TubeFormer [20]. Our Video-kMaX is built with the official code-base [41]. Closely following the prior works [20, 42], both the near-online and online models employ a specific pre-training protocol for KITTI-STEP and VIPSeg. They all commonly require ImageNet [33] pretrained checkpoint. VIPSeg further requires pre-training models on COCO [27]. For KITTI-STEP, Cityscapes [10] is additionally adopted as a pre-training dataset since they share a similar driving scene and class category. We note that our best backbone ConvNeXt-L [28] in KITTI-STEP uses both COCO [27] and Cityscapes [10] for pre-training model.

4.3. Main Results

KITTI-STEP Tab. 1 summarizes our performance on the KITTI-STEP *val* and test sets. On the validation set (Tab. 1 (a)), we compare methods in the two categories: online and near-online methods. In the online setting, when using the standard ResNet50 [14], our Video-kMaX (online) outperforms Video K-Net [24] by **+2.5%** STQ. To further push the envelope, our model, equipped with the modern backbone ConvNeXt-L [28], achieves the new state-of-the-art with 76.5% STQ. In the near-online setting, when using ResNet50, our Video-kMaX (near-online) significantly surpasses Motion-DeepLab [42] by **+16.2%** STQ. When employing Axial-ResNet50-B1 [39] backbone, Video-kMaX (near-online) also outperforms TubeFormer [20] by **+2.8%** STQ. Finally, Video-kMaX (near-online) with ConvNeXt-L further sets a new state-of-the-art performance with 78.9% STQ, significantly outperforming current best result (TubeFormer with Axial-ResNet50-B3) by **+4.6%** STQ. We observe the same trend on the test set (Tab. 1 (b)), where our model reaches 68.5% STQ, significantly outperforming the prior arts TubeFormer [20], Video K-Net [24], and Motion-DeepLab [42] by **+3.2%**, **+5.5%**, and **+16.3%** STQ, respectively. Remarkably, our extremely simple model even outperforms the ICCV 2021 Challenge winning entry, UW_IPL/ETRI_AIRL [54] by **+0.9%** STQ, which exploits pseudo labels [4, 59] and adopts an exceedingly complicated system that not only consists of separate tracking, detection, and segmentation modules, but also requires 3D object and depth information.

VIPSeg Tab. 2 (a) summarizes the results on the VIPSeg *val* set. In the online setting, our Video-kMaX (online) with ResNet50 attains 38.7% STQ / 36.8% VPQ, significantly outperforming the prior art Video K-Net by **+5.6% STQ / +10.7%** VPQ. Using the ConvNeXt-L backbone, our model advances the new state-of-the-art to 49.4% STQ / 49.4% VPQ. In the near-online setting, when using ResNet50, our Video-kMaX (near-online) surpasses Clip-PanoFCN [30] by **+8.4%** STQ / **+15.3%** VPQ. When using Axial-ResNet50-B1, Video-kMaX (near-online) outperforms TubeFormer [20] by **+6.0%** STQ / **+17.5%** VPQ. Our best setting with ConvNeXt-L backbone further advances the state-of-the-art to 51.7% STQ / 51.9% VPQ, outperforming TubeFormer with Axial-ResNet50-B3 by **+10.2%** STQ / **+20.7%** VPQ. We also show the effectiveness of Video-kMaX (near-online) on VIPSeg *test* set in Tab. 2 (b), where Video-kMaX also sets a new state-of-the-art, outperforming TubeFormer [20] by **+8.5%** STQ / **+18.2%** VPQ.

4.4. Ablation Studies

Association Modules Our proposed LA-MB exploits (1) Video Stitching (VS), (2) appearance feature, and (3) location feature, to perform the object association. In Tab. 3, we carefully study the effect of each feature in LA-MB under both near-online and online settings . In the near-online setting (Tab. 3 (a)), when using these three features *individually*, we discover that both VS and location feature are equally more effective than appearance feature. We note that when using only the appearance, the method becomes the naïve-MB approach, used by other works [16,45]. Combining all of them leads to our best final setting, while taking out the location feature will degrade the AQ performance most. This study demonstrates that our proposed location feature is the most effective feature among them. In the online setting, since the VS strategy becomes infeasible, we only experiment with the appearance and location features. As shown in Tab. 3 (b), the pure image-based model, which does not exploit any association feature, attains the worst performance. Interestingly, we notice that the appearance feature learned by the ResNet50 [14] is less effective than ConvNeXt-L [28]. When the appearance feature is less effective (*e.g*., when using ResNet50), it is better to just use the location feature for association. On the other hand, when the appearance feature is sufficiently informative (*e.g*., when using ConvNeXt-L), the best performance is obtained by using both appearance and location features.

method	association features: video-stitching	appearance	location	AQ
Video-kMaX (near-online)	✓			72.3
		✓		71.4
			✓	72.3
		✓	✓	73.8
	✓	✓		72.1
	✓		✓	73.6
	✓	✓	✓	**74.2**

(a) Near-online setting using clip-based trained models.

method	backbone	association features: appearance	location	AQ
Video-kMaX (online)	ResNet50			10.0
		✓		33.8
			✓	**72.0**
		✓	✓	66.4
	ConvNeXt-L			10.4
		✓		61.6
			✓	74.0
		✓	✓	**75.7**

(b) Online setting using image-based trained models.

Table 3. Ablation study on **different association features**, including the Video Stitching strategy, appearance feature, and location feature, on KITTI-STEP *val* set. We note that employing different association features will only affect the association quality (AQ). Our final LA-MB setting is labeled with brown color, while video-stitching and naïve-MB baselines are denoted in blue and red, respectively.

Memory-related Hyper-parameters Our proposed memory module LA-MB contains two hyper-parameters: τ (for refreshing old objects in the memory buffer) and α (confidence threshold for matching). In Tab. 4, we ablate their effects on our LA-MB and the baseline naïve-MB. As shown in the table, our LA-MB not only performs better, but also is more robust to the hyper-parameter values than naïve-MB. More concretely, when computing the mean and standard deviation (std) for the obtained AQ w.r.t. different τ and α, our LA-MB achieves a mean of 73.4 and a std of 0.4, while the baseline naïve-MB attains a *lower* mean of 61.5 and a *higher* std of 5.8. We think the robustness of LA-MB could be attributed to its efficient hierarchical matching scheme, which avoids the ambiguity caused by the large matching space.

Feature-related Hyper-parameter We adopt a temperature T to scale the values between the location and appearance features (see Eq. (5)). As shown in Tab. 5, our model is robust to the different values of T. We thus default its value to 1 for simplicity. Additionally, as shown in Tab. 6, our model is also robust to the different values of λ, which balances the weight between the stored appearance

AQ (%)	hyper-parameter set [τ / α]				row-wise (mean / std)
	[1 / 0.6]	[3 / 0.6]	[10 / 0.6]	[20 / 0.6]	
naïve-MB	69.5	67.1	54.3	47.4	59.6 / 10.5
LA-MB	72.7	73.0	73.9	73.7	73.3 / 0.6
	[1 / 0.7]	[3 / 0.7]	[10 / 0.7]	[20 / 0.7]	
naïve-MB	69.9	68.8	57.4	50.7	61.7 / 9.3
LA-MB	72.8	73.4	74.2	74.2	73.6 / 0.7
	[1 / 0.8]	[3 / 0.8]	[10 / 0.8]	[20 / 0.8]	
naïve-MB	71.4	71.3	64.9	59.5	66.8 / 5.7
LA-MB	72.6	73.2	73.6	73.6	73.3 / 0.5
column-wise (mean / std)	69.7 / 1.0 72.7 / 0.1	68.0 / 2.1 73.2 / 0.2	55.8 / 5.5 74.1 / 0.3	49.1 / 6.3 74.0 / 0.3	61.5 / 5.8 73.4 / 0.4

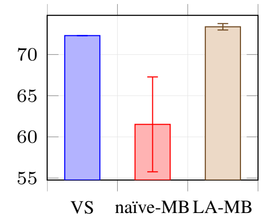
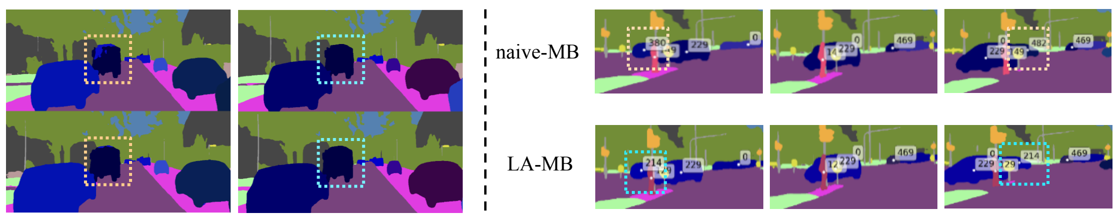


Table 4. Ablation study on **stability of Video-kMaX using different memory-related hyper-parameter sets (τ for memory-refreshing and α for confidence threshold)** on KITTI-STEP *val* set. We vary $\tau \in \{1, 3, 10, 20\}$ (different columns in the table) and $\alpha \in \{0.6, 0.7, 0.8\}$ (different rows in the table). We compute the mean and standard deviation column-wise (fixed τ and varied α), row-wise (varied τ and fixed α), and table-wise (varied τ and α). We plot the mean and standard deviation for the whole table on the right. The proposed LA-MB is more robust to the hyper-parameter values than the naïve-MB approach. Our final LA-MB setting and the naïve-MB baseline are labeled with brown and red color, respectively.



(a) TubeFormer (b) clip-kMaX (c) clip-kMaX + different memory buffer scheme

Figure 5. **Visualization results** on KITTI-STEP *val* set. The proposed within-clip segmenter, clip-kMaX, segments objects in a clip better than the state-of-art TubeFormer ((a) *vs.* (b)). In (c), the proposed cross-clip associater, LA-MB (Location-Aware Memory Buffer), associates occluded objects better than the baseline naïve-MB, which exploits only appearance features.

method	T	AQ	STQ
Video-kMaX (near-online)	0.5	73.95	74.10
	1.0	74.22	74.23
	1.5	74.30	74.27

Table 5. Ablation study on **temperature** T, which scales the values between location and appearance features. Our final setting is labeled with gray color. In this table, we show results up to two decimal points to more clearly see the robustness to T.

method	λ	AQ	STQ
Video-kMaX (near-online)	0.0	73.34	73.80
	0.5	74.19	74.21
	0.7	74.22	74.23
	0.8	74.22	74.23
	0.9	74.22	74.23
	1.0	74.18	74.21

Table 6. Additional analysis on **moving average weight** λ, which balances the stored appearance feature in the memory and current appearance feature. Our final setting is labeled with gray color. In this table, we show results up to two decimal points to more clearly see the robustness to λ.

feature in memory and the current one (see Eq. (3)).

Visualization Analysis We visualize results in Fig. 5 for KITTI-STEP. clip-kMaX performs better than the state-of-the-art TubeFormer [20] for consistent segmentation between frames in a clip. The proposed LA-MB enables long-term association, successfully re-identifying the occluded car object (ID 214), while the baseline naïve-MB fails, since it only exploits the appearance feature.

5. Conclusion

In this work, we proposed Video-kMaX, a unified framework for online and near-online Video Panoptic Segmentation (VPS) model with two modules: clip-kMaX and LA-MB. The clip-kMaX utilizes object queries as cluster centers to group pixels of the same object within a clip, while the LA-MB is a novel and robust memory module for both short- and long-term association with a hierarchical matching scheme. The effectiveness of our approach is demonstrated on the KITTI-STEP and VIPSeg datasets.

Acknowledgment

This work was supported by the NRF (NRF-2020M3H8A1115028, FY2021).

References

[1] Gedas Bertasius and Lorenzo Torresani. Classifying, segmenting, and tracking object instances in video with mask propagation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[2] Jiarui Cai, Mingze Xu, Wei Li, Yuanjun Xiong, Wei Xia, Zhuowen Tu, and Stefano Soatto. Memot: Multi-object tracking with memory. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8090–8100, 2022.

[3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Proceedings of the European Conference on Computer Vision*, pages 213–229. Springer, 2020.

[4] Liang-Chieh Chen, Raphael Gontijo Lopes, Bowen Cheng, Maxwell D Collins, Ekin D Cubuk, Barret Zoph, Hartwig Adam, and Jonathon Shlens. Naive-Student: Leveraging Semi-Supervised Learning in Video Sequences for Urban Scene Segmentation. In *Proceedings of the European Conference on Computer Vision*, 2020.

[5] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):834–848, 2017.

[6] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European Conference on Computer Vision*, pages 801–818, 2018.

[7] Bowen Cheng, Anwesa Choudhuri, Ishan Misra, Alexander Kirillov, Rohit Girdhar, and Alexander G Schwing. Mask2former for video instance segmentation. *arXiv:2112.10764*, 2021.

[8] Bowen Cheng, Maxwell D Collins, Yukun Zhu, Ting Liu, Thomas S Huang, Hartwig Adam, and Liang-Chieh Chen. Panoptic-DeepLab. In *ICCV COCO + Mapillary Joint Recognition Challenge Workshop*, 2019.

[9] Bowen Cheng, Maxwell D Collins, Yukun Zhu, Ting Liu, Thomas S Huang, Hartwig Adam, and Liang-Chieh Chen. Panoptic-DeepLab: A simple, strong, and fast baseline for bottom-up panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[10] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3213–3223, 2016.

[11] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of IEEE International Conference on Computer Vision*, 2017.

[12] Raghudeep Gadde, Varun Jampani, and Peter V Gehler. Semantic video CNNs through representation warping. In *Proceedings of IEEE International Conference on Computer Vision*, 2017.

[13] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of IEEE International Conference on Computer Vision*, 2017.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[15] Miran Heo, Sukjun Hwang, Seoung Wug Oh, Joon-Young Lee, and Seon Joo Kim. Vita: Video instance segmentation via object token association. *arXiv:2206.04403*, 2022.

[16] De-An Huang, Zhiding Yu, and Anima Anandkumar. Minvis: A minimal video instance segmentation framework without video-based training. *Advances in Neural Information Processing Systems*, 2022.

[17] Sukjun Hwang, Miran Heo, Seoung Wug Oh, and Seon Joo Kim. Video instance segmentation using inter-frame communication transformers. *Advances in Neural Information Processing Systems*, 2021.

[18] Samvit Jain, Xin Wang, and Joseph 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.

[19] Dahun Kim, Sanghyun Woo, Joon-Young Lee, and In So Kweon. Video panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9859–9868, 2020.

[20] Dahun Kim, Jun Xie, Huiyu Wang, Siyuan Qiao, Qihang Yu, Hong-Seok Kim, Hartwig Adam, In So Kweon, and Liang-Chieh Chen. Tubeformer-deeplab: Video mask transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13914–13924, 2022.

[21] Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollár. Panoptic feature pyramid networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

[22] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

[23] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

[24] Xiangtai Li, Wenwei Zhang, Jiangmiao Pang, Kai Chen, Guangliang Cheng, Yunhai Tong, and Chen Change Loy. Video k-net: A simple, strong, and unified baseline for video segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18847–18857, 2022.

[25] Yanwei Li, Xinze Chen, Zheng Zhu, Lingxi Xie, Guan Huang, Dalong Du, and Xingang Wang. Attention-guided unified network for panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

[26] Yanwei Li, Hengshuang Zhao, Xiaojuan Qi, Liwei Wang, Zeming Li, Jian Sun, and Jiaya Jia. Fully convolutional

networks for panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

[27] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Proceedings of the European Conference on Computer Vision*, 2014.

[28] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11976–11986, 2022.

[29] Tim Meinhardt, Alexander Kirillov, Laura Leal-Taixe, and Christoph Feichtenhofer. Trackformer: Multi-object tracking with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8844–8854, 2022.

[30] Jiaxu Miao, Xiaohan Wang, Yu Wu, Wei Li, Xu Zhang, Yunchao Wei, and Yi Yang. Large-scale video panoptic segmentation in the wild: A benchmark. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2022.

[31] Jiaxu Miao, Yunchao Wei, Yu Wu, Chen Liang, Guangrui Li, and Yi Yang. Vspw: A large-scale dataset for video scene parsing in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

[32] Siyuan Qiao, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Vip-deeplab: Learning visual perception with depth-aware video panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

[33] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[34] Peize Sun, Jinkun Cao, Yi Jiang, Rufeng Zhang, Enze Xie, Zehuan Yuan, Changhu Wang, and Ping Luo. Transtrack: Multiple object tracking with transformer. *arXiv:2012.15460*, 2020.

[35] Zhi Tian, Chunhua Shen, and Hao Chen. Conditional convolutions for instance segmentation. In *Proceedings of the European Conference on Computer Vision*, 2020.

[36] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 2017.

[37] Paul Voigtlaender, Michael Krause, Aljosa Osep, Jonathon Luiten, Berin Balachandar Gnana Sekar, Andreas Geiger, and Bastian Leibe. Mots: Multi-object tracking and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

[38] Huiyu Wang, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Max-deeplab: End-to-end panoptic segmentation with mask transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5463–5474, 2021.

[39] Huiyu Wang, Yukun Zhu, Bradley Green, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Axial-deeplab: Standalone axial-attention for panoptic segmentation. In *Proceedings of the European Conference on Computer Vision*, 2020.

[40] Xinlong Wang, Rufeng Zhang, Tao Kong, Lei Li, and Chunhua Shen. Solov2: Dynamic and fast instance segmentation. *Advances in Neural information processing systems*, 2020.

[41] Mark Weber, Huiyu Wang, Siyuan Qiao, Jun Xie, Maxwell D Collins, Yukun Zhu, Liangzhe Yuan, Dahun Kim, Qihang Yu, Daniel Cremers, et al. Deeplab2: A tensorflow library for deep labeling. *arXiv preprint arXiv:2106.09748*, 2021.

[42] Mark Weber, Jun Xie, Maxwell Collins, Yukun Zhu, Paul Voigtlaender, Hartwig Adam, Bradley Green, Andreas Geiger, Bastian Leibe, Daniel Cremers, Aljosa Osep, Laura Leal-Taixe, and Liang-Chieh Chen. Step: Segmenting and tracking every pixel. *Neural Information Processing Systems (NeurIPS) Track on Datasets and Benchmarks*, 2021.

[43] Sanghyun Woo, Dahun Kim, Joon-Young Lee, and In So Kweon. Learning to associate every segment for video panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2705–2714, 2021.

[44] Junfeng Wu, Yi Jiang, Song Bai, Wenqing Zhang, and Xiang Bai. Seqformer: Sequential transformer for video instance segmentation. In *Proceedings of the European Conference on Computer Vision*, pages 553–569. Springer, 2022.

[45] Junfeng Wu, Qihao Liu, Yi Jiang, Song Bai, Alan Yuille, and Xiang Bai. In defense of online models for video instance segmentation. In *Proceedings of the European Conference on Computer Vision*, pages 588–605. Springer, 2022.

[46] Jialian Wu, Sudhir Yarram, Hui Liang, Tian Lan, Junsong Yuan, Jayan Eledath, and Gerard Medioni. Efficient video instance segmentation via tracklet query and proposal. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 959–968, 2022.

[47] Yuwen Xiong, Renjie Liao, Hengshuang Zhao, Rui Hu, Min Bai, and Raquel Urtasun Ersin Yumer. Upsnet: A unified panoptic segmentation network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

[48] Yuwen Xiong, Renjie Liao, Hengshuang Zhao, Rui Hu, Min Bai, Ersin Yumer, and Raquel Urtasun. UPSNet: A unified panoptic segmentation network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

[49] Linjie Yang, Yuchen Fan, and Ning Xu. Video Instance Segmentation. In *Proceedings of IEEE International Conference on Computer Vision*, 2019.

[50] Tien-Ju Yang, Maxwell D Collins, Yukun Zhu, Jyh-Jing Hwang, Ting Liu, Xiao Zhang, Vivienne Sze, George Papandreou, and Liang-Chieh Chen. DeeperLab: Single-shot image parser. *arXiv:1902.05093*, 2019.

[51] Qihang Yu, Huiyu Wang, Dahun Kim, Siyuan Qiao, Maxwell Collins, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Cmt-deeplab: Clustering mask transformers for panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[52] Qihang Yu, Huiyu Wang, Siyuan Qiao, Maxwell Collins, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. k-means Mask Transformer. In *Proceedings of the European Conference on Computer Vision*, pages 288–307. Springer, 2022.

[53] Fangao Zeng, Bin Dong, Yuang Zhang, Tiancai Wang, Xiangyu Zhang, and Yichen Wei. Motr: End-to-end multiple-object tracking with transformer. In *Proceedings of the European Conference on Computer Vision*, pages 659–675. Springer, 2022.

[54] Haotian Zhang, Yizhou Wang, Zhongyu Jiang, Cheng-Yen Yang, Jie Mei, Jiarui Cai, Jenq-Neng Hwang, Kwang-Ju Kim, and Pyong-Kun Kim. U3D-MOLTS: Unified 3D Monocular Object Localization, Tracking and Segmentation. In *ICCV Segmenting and Tracking Every Point and Pixel: 6th Workshop on Benchmarking Multi-Target Tracking*, 2021.

[55] Wenwei Zhang, Jiangmiao Pang, Kai Chen, and Chen Change Loy. K-net: Towards unified image segmentation. *Advances in Neural Information Processing Systems*, 34:10326–10338, 2021.

[56] Zelin Zhao, Ze Wu, Yueqing Zhuang, Boxun Li, and Jiaya Jia. Tracking objects as pixel-wise distributions. In *Proceedings of the European Conference on Computer Vision*, pages 76–94. Springer, 2022.

[57] Yi Zhou, Hui Zhang, Hana Lee, Shuyang Sun, Pingjun Li, Yangguang Zhu, ByungIn Yoo, Xiaojuan Qi, and Jae-Joon Han. Slot-vps: Object-centric representation learning for video panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3093–3103, 2022.

[58] Xizhou Zhu, Yuwen Xiong, Jifeng Dai, Lu Yuan, and Yichen Wei. Deep feature flow for video recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2017.

[59] Yi Zhu, Karan Sapra, Fitsum A Reda, Kevin J Shih, Shawn Newsam, Andrew Tao, and Bryan Catanzaro. Improving semantic segmentation via video propagation and label relaxation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.