Video K-Net: A Simple, Strong, and Unified Baseline for Video Segmentation

Xiangtai Li[1*], Wenwei Zhang[2*], Jiangmiao Pang[3,5*], Kai Chen[4,5], Guangliang Cheng[4], Yunhai Tong[1], Chen Change Loy[2]

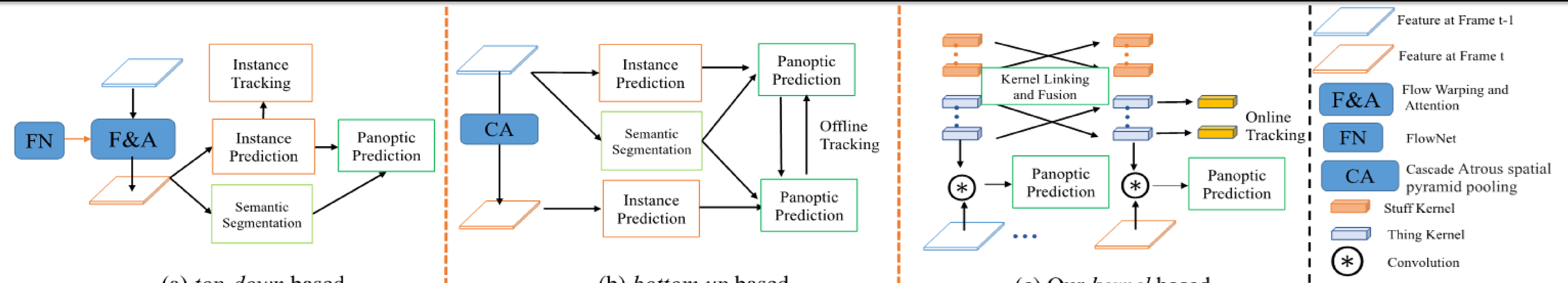
[1]Peking University, [2]S-Lab, Nanyang Technological University, [3]The Chinese University of Hong Kong, [4]SenseTime Research, [5]Shanghai AI Laboratory

1. Motivation

Summary of Video Segmentation Tasks:

- Different video segmentation tasks have **different** solutions.
- Video Semantic Segmentation (VSS): no instance tracking.
- Video Instance Segmentation (VIS): no background context.
- Video Panoptic Segmentation (VPS): unifies VSS and VIS.
- **Is there a unified solution to handle all three tasks?**

Figure From: A Survey on Deep Learning Technique for Video Segmentation



Current Solution For Video Panoptic Segmentation

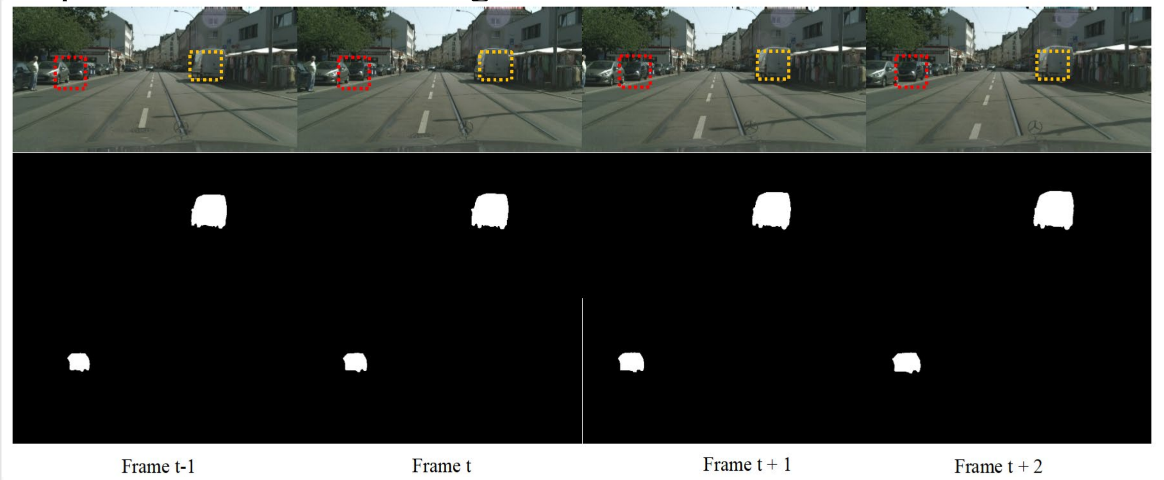
- Complex and hand-crafted pipelines vary from models (a: top-down, b: bottom up).
- Tackle the segmentation and tracking with specific task heads (a, b): semantic/instance/tracking heads
- Need post process and offline tracking (b) or extra optical flow learning and warping (a).
- **Is there a simple solution to handle VPS problem?**

Key Motivation:

- Image segmentation tasks are *already unified* by kernel based method like **K-Net.**
- Kernel based method can also simplify video panoptic segmentation. → **Unify Video Segmentation Tasks via Kernels**
- Adopting Kernel based method can generalize into VSS and VIS.

2. Toy Experiment

Yellow boxes: kernel-2, red boxes: kernel-3. The last two rows represent masks according to the kernels.



KITTI-STEP	STQ	AQ	SQ	VPQ
K-Net	67.5	65.5	68.9	-
K-Net + Unitrack	65.1	64.3	68.9	-
Cityscapes-VPS	STQ	AQ	SQ	VPQ
K-Net	-	-	-	54.3
K-Net + Unitrack	-	-	-	53.2

K-Net is better than K-Net + Unitrack!

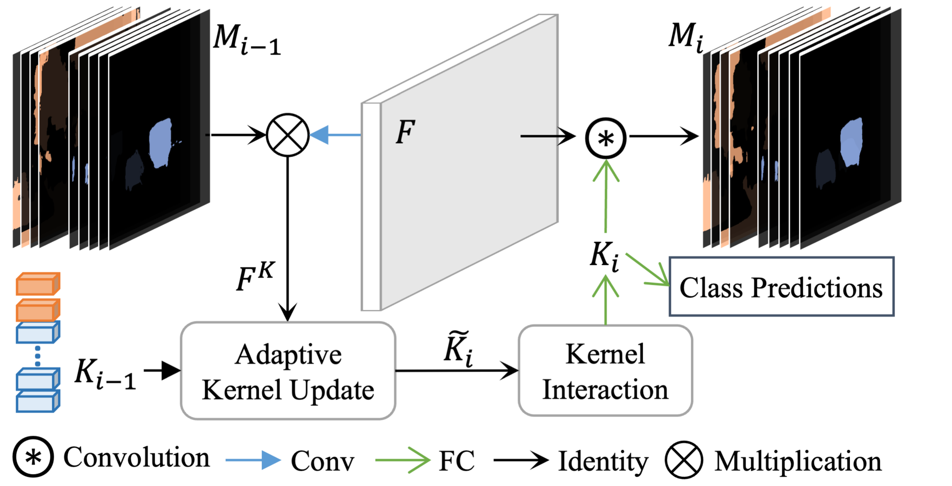
Observation

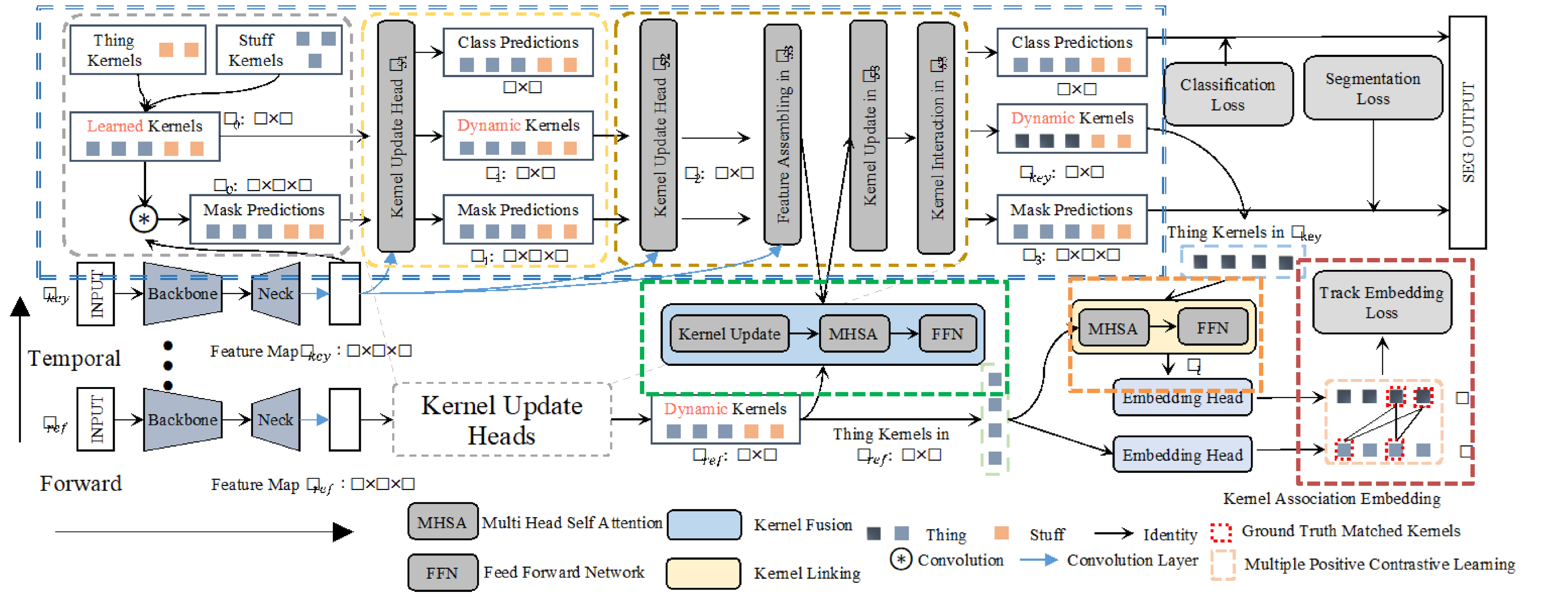
- Original learned kernels encode instance-wised information.
- Directly using kernel can lead to better performance than several advanced trackers.
- **Let's just link and track the kernels in temporal dimension!**

3. Method

Original K-Net (blue dash box in below figure)

- Both thing kernels and stuff kernels are randomly initialized and learns to segment
- Kernel Update Head update kernels by:
  - Group Feature Assembling
  - Adaptive Kernel Update
  - and Kernel Interaction sequentially





Video K-Net: Extending K-Net into Video Domain

- Three modifications based on the image K-Net.
- Only **one** extra tracking loss into K-Net.
- Online Inference *without* offline posting processing.

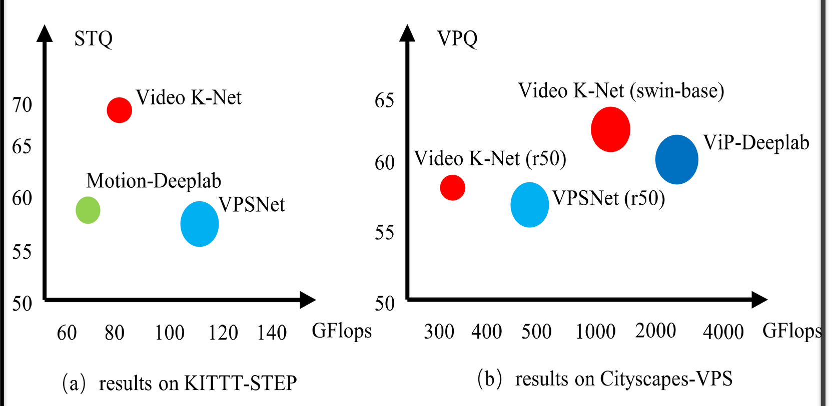


Figure 1. The best performance and GFlops trade-off for both KITTI-STEP (a) and Cityscapes VPS (b).

Lighter and Stronger

Modification 1: Learning to Fuse Kernels (green box)

- Fuse the kernel at the last kernel update stage.
- Use Adaptive Kernel Update (operation from K-Net).

Modification 2: Learning to Link Kernels (orange box)

- Message passing from reference frames key frames.
- Link thing kernels across temporal dimension via Multi Head Self Attention (MHSA).

Modification 3: Learning Kernel Association Embeddings (red box)

- Apply mask-based assignment to associate kernels and masks.
- Adopt **sparse** kernel association rather than quasi-dense regional proposals.
- **v** kernels in key frame are matched with **k** kernels ($\mathbf{k}^+$ positative, $\mathbf{k}^-$ negative) in reference frames via a temporal contrastive loss $\mathbf{L_{track}}$:

$$\mathbf{L_{track}} = -\sum_{\mathbf{k}^+} \log \frac{\exp(\mathbf{v}\cdot\mathbf{k}^+)}{\exp(\mathbf{v}\cdot\mathbf{k}^+) + \sum_{\mathbf{k}^-}\exp(\mathbf{v}\cdot\mathbf{k}^-)}$$

4. Experiments

Table 1. Ablation Studies on KITTI-STEP(%).

KAE	KL	KF	STQ	AQ	SQ
×	×	×	67.5	65.5	68.9
√	×	×	69.3	69.0	69.8
√	√	×	70.2	71.2	69.7
√	√	√	70.9	70.8	71.2

- Baseline: original K-Net, M: Modification
- KAE: Kernel Association Embedding (M3)
- KL: Kernel Linking (M2) KF: Kernel Fusion (M1)

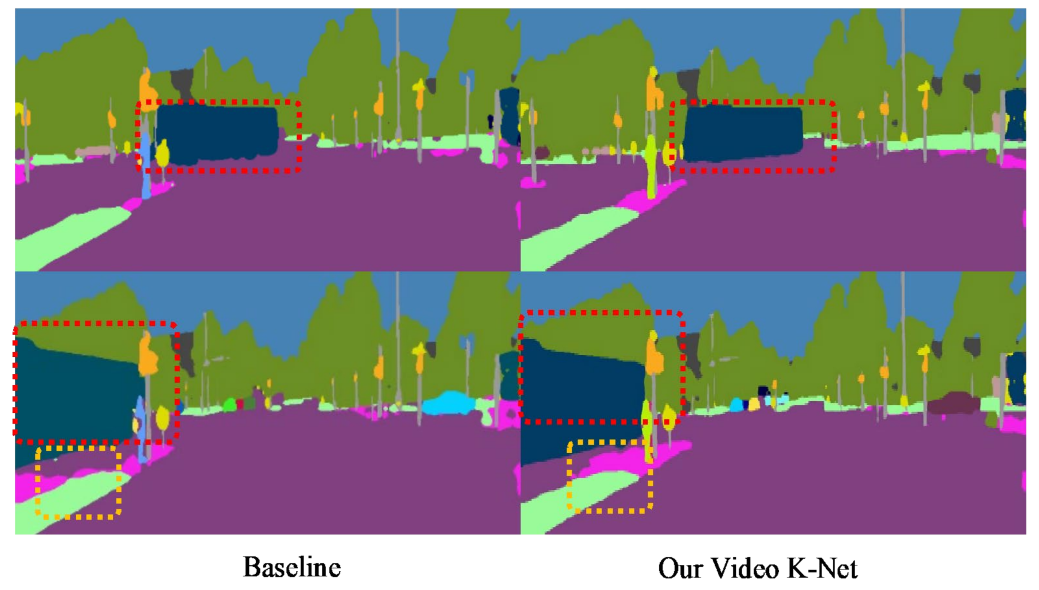


Figure 2. Visual Improvements over K-Net baseline.

Table 2. Results on KITTI-STEP using ResNet-50 backbone.

Method	STQ	AQ	SQ	VPQ
P + SORT	0.59	0.50	0.71	0.42
P+ Mask Propagation	0.67	0.63	0.71	0.44
P +IoU Assoc	0.58	0.47	0.71	0.44
Motion-Deeplab	0.58	0.51	0.67	0.40
VPSNet	0.56	0.52	0.61	0.43
Video K-Net	0.71	0.70	0.71	0.46

- Video K-Net achieves 3.4% improvements over strong K-Net baseline for STQ metric.
- Video K-Net achieves relatively **10% improvements** over Motion Deeplab on KITTI-STEP for STQ metric.
- Better performance with less GFlops than previous top-down and bottom up methods on Cityscapes VPS.
- Generalizes well on VIS (Youtube-2019) and VSS (VSPW).

Table 3. Results on Youtube-VIS using ResNet-50 backbone.

Method	AP	AP_{50}	AP_{75}	AR_1	AR_{10}
MaskProp	40.0	-	42.9	-	-
STEm-Seg	30.6	50.7	33.5	31.6	37.1
CompFeat	35.3	56.0	38.6	33.1	40.3
VISTR	34.4	55.7	36.5	33.5	40.4
Video K-Net	40.5	63.5	44.5	40.7	49.9

Table 4. Results on VSPW-VSS using ResNet-101 backbone.

Method	mIoU	mVC_8	mVC_{16}
Deeplav3+	35.7	83.5	78.4
PSPNet	36.5	84.4	79.8
TCB (PSPNet)	37.5	86.9	82.1
Video K-Net (PSPNet)	37.9	87.0	82.1
Video K-Net (Deeplabv3+)	38.0	87.2	82.3

Figure 3. Visual Results on Cityscapes VPS (left) and KITTI STEP (right).