A1. Details of the spatio-temporal fusion block

We design a fusion block specifically for spatio-temporal applications, namely Spatio-Temporal Attention Module (STAM), as discussed in Sec. 3.1. STAM is based on the attention mechanism inspired by [48]. Consider the concatenation of the propagated previous feature z'_{t-1} and the current feature z_t as $z'_{(t-1,t)} \in \mathbb{R}^{T\times C\times H\times W}$, the STAM process can be written as:

$$\begin{aligned} z_{(t-1,t)} = \{[A_{spa}[A_{tem}(z'_{(t-1,t)}) \otimes z'_{(t-1,t)}] \\ \otimes [A_{tem}(z'_{(t-1,t)}) \otimes z'_{(t-1,t)}]\} \oplus z'_{(t-1,t)}, \end{aligned} \quad (5)$$

where A_{tem} is temporal attension, A_{spa} is spatial attension, $\otimes$ denotes element-wise multiplication, and $\oplus$ denotes element-wise addition.

Temporal attention. The proposed temporal attention mechanism learns to choose informative temporal elements along each pixel's temporal dimension in the spatio-temporal space. The temporal attention $A_{tem} \in \mathbb{R}^{T\times 1\times 1\times 1}$ is performed as:

$$A_{tem}(z) = \sigma(FC(AvgPool(z)) + FC(MaxPool(z))), \quad (6)$$

where σ is the sigmoid function, and FC denotes a fully connected layer.

Spatial attention. The spatial attention mechanism chooses informative pixels along the spatial dimension in the spatio-temporal space. The spatial attention $A_{spa} \in \mathbb{R}^{1\times 1\times H\times W}$ is performed as:

$$A_{spa}(z) = \sigma(Conv(Concat[AvgPool(z), MaxPool(z)])), \quad (7)$$

where $Concat$ denotes the concatenation operation, and $Conv$ denotes a convolutional layer.

Remark. The main contribution of this paper is the STPL framework. In Table 4, we can see that STPL can outperform all the existing methods even with the very simple Concatenation fusion, showing its flexibility. We propose STAM to show that STPL can further benefit from a more advanced fusion module.

A2. Temporal consistency

We quantitatively compare the temporal consistency of different objective functions. The temporal consistency is derived from the overlap between the predicted segmentation maps of successive frames. We compute the percentage of the overlapping pixels. As shown in Table 5, STPL performs the best, indicating that the proposed spatio-temporal method significantly improves temporal consistency. This quantitative result is consistent with the qualitative results shown in Figure 5.

Table 5. Temporal consistency of different objective functions on VIPER [36] → Cityscapes-Seq [7].

Method / Objective function	Consistency (%)
Source-only	72.93
Temporal-only CL ($\mathcal{L}^{tem}$)	75.84 (+2.91)
Spatial-only CL ($\mathcal{L}^{spa}$)	77.68 (+4.75)
Naïve T+S CL ($\mathcal{L}^{tem} + \mathcal{L}^{spa}$)	80.91 (+7.89)
STPL (Ours; $\mathcal{L}^{stpl}$)	**82.14 (+9.21)**

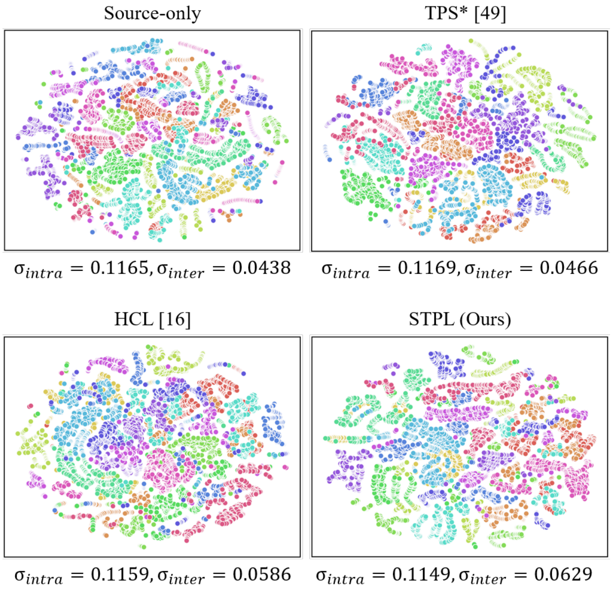

Figure 8. The t-SNE visualization [42] of the feature space learned for VIPER [36] → Cityscapes-Seq [7], where each point in the scatter plots stands for a pixel representation. All 15 classes are sampled to visualize. σ_{intra} is the intra-class variance (lower is better) and σ_{inter} is the inter-class variance (higher is better) of the feature space. All the methods are evaluated on the same selected video samples. In comparison, the proposed STPL learns the most discriminative feature space, which is reflected by the lowest σ_{intra} and the highest σ_{inter}.

A3. More on feature visualization

Figure 6 provides the t-SNE visualization [42] of the feature space learned for the VIPER → Cityscapes-Seq benchmark, where only four classes are sampled for simplicity. In this section, we visualize all 15 classes (see Figure 8). As can be seen, the proposed STPL learns the most discriminative feature space. It acquires the lowest σ_{intra} and the highest σ_{inter}. This once again demonstrates STPL's ability to learn semantic correlations among pixels in the spatio-temporal space.