A. Technical Appendices and Supplementary Material

A.1. DiffAct: Diffusion Action Segmentation

DiffAct [48] introduces a novel generative approach for temporal action segmentation by leveraging denoising diffusion probabilistic models (DDPMs). Unlike prior methods that operate deterministically, DiffAct formulates action segmentation as a conditional generation problem, where frame-wise action sequences are generated from pure noise conditioned on video features. In this paper, we adopt DiffAct's model architecture and input masking strategies during training.

Diffusion-based formulation. Given an input video with L frames and corresponding ground truth one-hot action labels $Y_0 \in \{0,1\}^{L \times C}$ (where C is the number of action classes), and an encoder h_ϕ. The encoder encodes the input video features $F \in \mathbb{R}^{L \times D}$ using $E = h_\phi(F)$. A decoder g_ψ is trained to denoise the noisy label sequence Y_t at timestep t conditioned on encoded features E, producing action logits $P_t \in \{0,1\}^{L \times C}$.

Training. Beyond proposing novel euclidean training objectives, DiffAct uses a condition masking strategy rooted in human behavior modelling. Specifically, they integrate three human action priors into the diffusion framework. Firstly, *No Masking*, which passes all features into the decoders. Secondly, *Masking for Position Prior* and *Masking for Boundary Prior* to enforce the model to rely only on frame positions and explore action boundaries. Lastly, *Masking for Relation Prior* prompts the model to infer the missing action segment.

Inference. The denoising decoder g_ψ is trained to handle inputs with varying levels of noise, even sequences composed entirely of random noise. During inference, the process begins with a purely noisy sequence $\hat{Y}_T \sim \mathcal{N}(0, I)$ and gradually removes the noise through an iterative denoising procedure. At each step t, the sequence is updated using:

$$\hat{Y}_{t-1} = \sqrt{\bar{\alpha}_{t-1}} P_t + \frac{\sqrt{1 - \bar{\alpha}_{t-1} - \sigma_s^2}}{\sqrt{1 - \bar{\alpha}_t}} (\hat{Y}_t - \sqrt{\bar{\alpha}_s} P_t) + \sigma_t \epsilon \quad (14)$$

where $\hat{Y}_{t-1}$ is passed into the decoder to produce the next prediction P_{t-1}. This process continues step-by-step, refining the noisy sequence $\hat{Y}_T, \hat{Y}_{T-1}, \ldots, \hat{Y}_0$ until the final output $\hat{Y}_0$, which closely approximates the true action sequence.

To accelerate inference, DiffAct adopts a sampling trajectory that skips intermediate steps, producing a shorter sequence such as $\hat{Y}_S, \hat{Y}_{S-\Delta}, \ldots, \hat{Y}_0$. Note that during inference, the encoded features E are fed into the decoder without any masking.

A.2. Background on Diffusion Models

Diffusion models learn to approximate a target data distribution by progressively corrupting data with Gaussian noise in a forward process, and then learning to reverse this corruption through a denoising neural network. The forward (or diffusion) process transforms clean data $\mathbf{x}_0$ into a noisy version $\mathbf{x}_t$ by gradually adding Gaussian noise according to a predefined variance schedule. Specifically, this process can be expressed as:

$$\mathbf{x}_t = \sqrt{\gamma(t)}\, \mathbf{x}_0 + \sqrt{1 - \gamma(t)}\, \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}) \quad (15)$$

where $\gamma(t)$ is a monotonically decreasing function that controls the noise magnitude at timestep $t \in \{1, 2, \ldots, T\}$.

In the reverse process, a neural network $f(\mathbf{x}_t, t)$ is trained to recover $\mathbf{x}_0$ from noisy inputs $\mathbf{x}_t$. This is typically done by minimizing a simple L2 reconstruction loss:

$$\mathcal{L} = \frac{1}{2} \left\| f(\mathbf{x}_t, t) - \mathbf{x}_0 \right\|_2^2 \quad (16)$$

At inference time, the model starts from a pure noise vector $\mathbf{x}_T$ and iteratively denoises it through the learned reverse trajectory $\mathbf{x}_T \rightarrow \mathbf{x}_{T-\Delta} \rightarrow \cdots \rightarrow \mathbf{x}_0$, ultimately reconstructing a sample from the original data distribution.

In our setting, the model learns to generate frame-wise action label sequences from Gaussian noise, conditioned on video features for action segmentation.

A.3. Additional Dataset

To validate our framework beyond cooking datasets, we utilize the YouTube Instructional (YTI) dataset [2]. The dataset consists of five tasks and thirty videos per task with an average video duration of two minutes. The data is coarsely labeled on 49 action categories. In Table 7, we evaluate DiffAct [2] and HybridTAS (Ours) on this dataset using the same evaluation metrics. Our proposed approach outperforms DiffAct across all metrics.

Method	F1@10	F1@25	F1@50	Edit	Acc	Avg
DiffAct [48]	53.4	45.5	27.5	56.5	71.1	50.8
HybridTAS (Ours)	**58.1**	**52.3**	**33.6**	**62.3**	**69.5**	**54.9**

Table 7. **Quantitative Results on the YTI dataset**.

B. Experiments

Datasets. We conduct experiments on three benchmark datasets: GTEA, 50Salads, and Breakfast. GTEA [23] consists of 28 egocentric videos of daily activities, covering

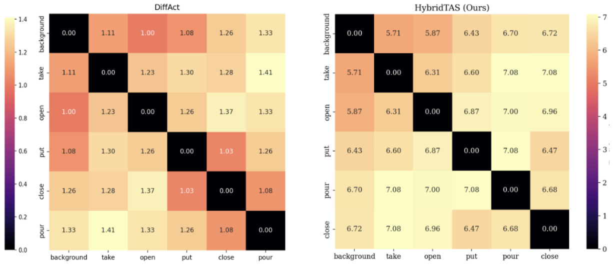

Figure 6. **Cluster centroid distances.** We plot the cluster centroid distances to showcase an almost 6x more distance in HybridTAS, which is indicative of better clustering. Note that the HybridTAS distances are hyperbolic distances, whereas DiffAct [48] distances are Euclidean.

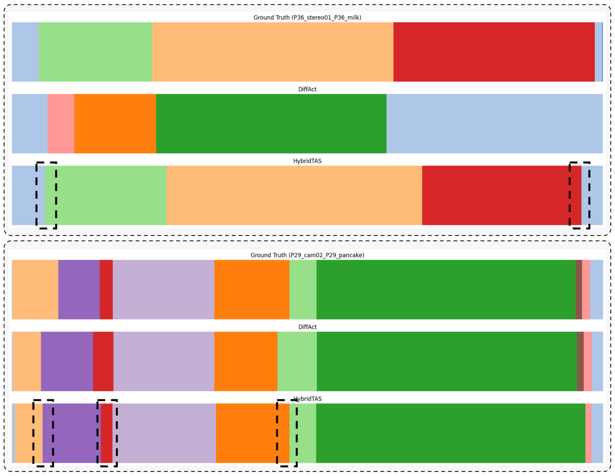

Figure 7. **Qualitative results on the Breakfast dataset [40].**We present a comparison of segmentation outputs of DiffAct [48] and HybridTAS (Ours) on *P30_stereo01_P36* (top) and *P29_cam02_P29_pancake* (bottom) with dashed boxes representing areas of improvement.

Hyperparamters	**50 Salads** [65]	**Breakfast** [40]	**GTEA** [23]
λ_{ce}	0.5	0.5	0.5
λ_{entail}	0.05	0.1	0.05
λ_{margin}	0.1	0.2	0.1
λ_{pp}	0.1	0.2	0.1
λ_{gg}	0.1	0.2	0.1
E_1	2000	400	4000
Curvature (c)	1.0	1.0	1.0
Total epochs	5000	1000	10000

Table 8. **Dataset specific hyperparamter values.**

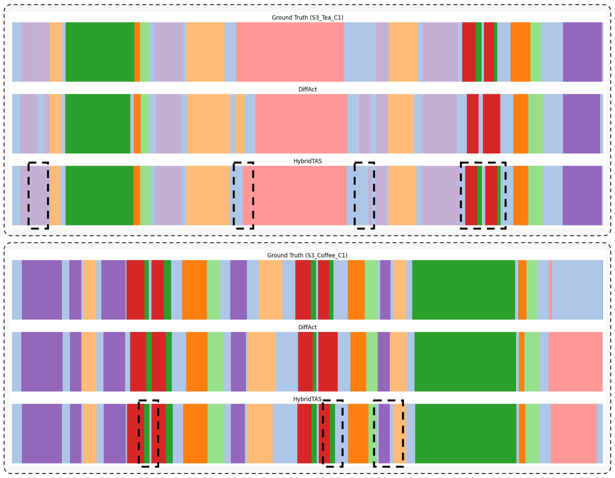

Figure 8. **Qualitative results on the GTEA dataset [23].**We present a comparison of segmentation outputs of DiffAct [48] and HybridTAS (Ours) on *S3_Tea_C1* (top) and *S3_Coffee_C1* (bottom) with dashed boxes representing areas of improvement.

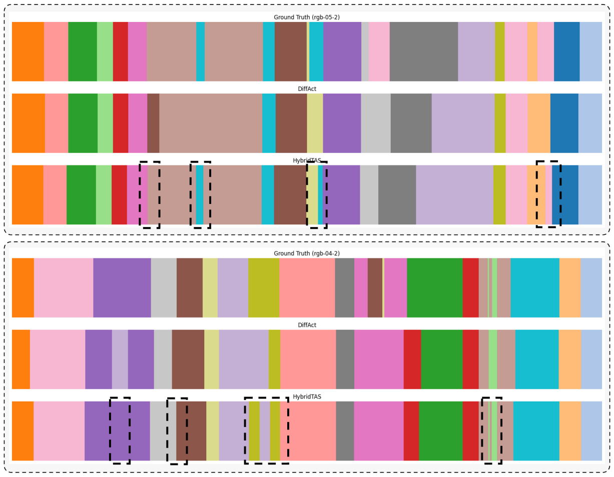

Figure 9. **Qualitative results on the 50Salads dataset [65].**We present a comparison of segmentation outputs of DiffAct [48] and HybridTAS (Ours) on *rgb-05-2* (top) and *rgb-04-2* (bottom) with dashed boxes representing areas of improvement.

11 action classes. Each video is approximately one minute long and contains around 19 action instances. 50Salads [65] features 50 top-view videos of salad preparation, annotated with 17 action classes. The videos average six minutes in length, with roughly 20 action instances per video. Breakfast [40] is a large-scale dataset comprising 1712 third-person videos spanning 48 action classes related to breakfast preparation. While the average video length is two minutes, there is significant variance across samples; each video contains around seven action instances on average. Among the three, Breakfast [40] offers the largest scale, while 50Salads [65] includes the longest videos and the highest number of instances per video. As in DiffAct [48], we adopt five-fold cross-validation on 50Salads and four-fold cross-validation on GTEA and Breakfast, using the same data splits for fair comparison.

Metrics. Following previous works [47, 74], the frame-wise accuracy (Acc), the edit score (Edit), and the F1 scores at overlap thresholds 10%, 25%, 50% (F1@10, 25, 50) are reported. The accuracy assesses the results at the frame level, while the edit score and F1 scores measure the performance at the segment level.

Implementation details. For all datasets, we utilize the I3D features [9] as the input features $\mathbf{F}$, whose dimension is 2048. The encoder h_ϕ and decoder g_ψ are adopted from DiffAct [48]. The encoder is a reimplementation of the ASFormer encoder [74], while the ASFormer decoder is modified to be step-aware by incorporating step embeddings into the input, as proposed in [32]. Specifically, the encoder contains 10, 10, 12 layers with 64, 64, 256 feature maps for the GTEA [23], 50Salads [65], and Breakfast [40] datasets. The decoder comprises of 8 layers with 24, 24, 128 feature maps for the respective datasets. Intermediate features from three encoder layers (5, 7, 9) are concatenated to be used as conditional input to the decoder. The entire framework is trained with the RiemannianAdam optimizer, a batch size of 4, a learning rate of $1e-4$ (Breakfast [40]) and $5e-4$ (GTEA [23] and 50Salads [65]. The total diffusion timesteps during training is set to $T = 1000$, and 25 steps are utilized during inference. We have performed all experiments on a single NVIDIA H100 GPU. Dataset-specific hyperparameters have been provided in Table 8.