Supplementary Material of Collaborative Static and Dynamic Vision-Language Streams for Spatio-Temporal Video Grounding

Zihang Lin[1], Chaolei Tan[1], Jian-Fang Hu[1,3,4]*, Zhi Jin[1], Tiancai Ye[2], Wei-Shi Zheng[1,3,4]
[1]Sun Yat-sen University, China [2]Tencent, China
[3]Guangdong Province Key Laboratory of Information Security Technology, China
[4] Key Laboratory of Machine Intelligence and Advanced Computing, Ministry of Education, China
{linzh59, tanchlei}@mail2.sysu.edu.cn, {hujf5, jinzh26}@mail.sysu.edu.cn,
tiancaiye@tencent.com, wszheng@ieee.org

In this supplementary material, we provide more details that are excluded from the main submission due to space limitations. We present more visualization results in Section A to better show the effectiveness of our approach, more implementation details are introduced in Section B and some additional experimental results are presented in Section C.

A. Visualized Results and Analysis

In this section, we provide more visualized results on HCSTVG-v2 validation set [S7] to show the rationality of our design and the effectiveness of the proposed method.

In Figure S1, we present a visualization of the attention map A^i which is employed in the static-to-dynamic information transmission block to guide the learning of the dynamic stream. As shown in the first two samples, the attention maps can attend to the objects that may match the query text, hence it can guide the dynamic stream to focus on motions in the region that may contain the target object. For the last sample, when the target man in the red stripe was not in the scene, the attention weights are low. When the target men appeared, the attention weights became much higher. The visualization of the attention maps shows the rationality of the design of our static-to-dynamic information transmission block.

In Figure S2, we show more comparison results with our baseline model without the cross-stream collaboration block. In the first two samples, the baseline model predicts the wrong temporal time span since it cannot understand the static visual cues described in the text like "red stripe" and "white clothes". With the proposed collaboration block, the dynamic stream can attend to the region that may contain the target object and neglect other regions (as shown in Figure S1), thus it can correctly predict the target time span. For the third sample, in the baseline model, the static stream failed to understand the concept of "blindfolded" and thus it produces incorrect spatial grounding prediction. By employing the proposed cross-stream collaboration block, the model can reason the target person according to the action "smell" so that it can accurately localize the target object. For the last sample, the baseline model can correctly localize the target person according to the description "at the door", but it fails to track the target man in the latter frames in which the door is out of the scene. In our model, the query mixing operation in the dynamic-to-static information transmission block helps the model track the target person. The above samples further demonstrate the effectiveness of the proposed cross-stream collaboration block.

In Figure S3, we show more prediction results of our approach, including both successful cases and failure cases. To show samples with different vIoU more intuitively so that we can get a better sense of the evaluation metric, here we select to present samples of vIoU around 0.8, 0.5 and 0.3 from HCSTVG-v2 validation set [S7]. As shown in the first sample in Figure S3, visually, our prediction is very close to the ground truth and the vIoU metric is 80.3%. In the second sample in which the vIoU is 50.2%, our predicted temporal boundary of action "sit down" is not that accurate while the predicted spatial bounding boxes are quite precise. In the last sample, the text query is long and it consists of a composition of several actions, which is quite challenging thus our model only achieves a vIoU of 29.1%. To show the visualization results better, we also provide a video-version visualization (the file named "visualization.mp4" in the same folder) in which we present several samples with different vIoUs (including all samples shown in Figure S3)
.

*Corresponding author.

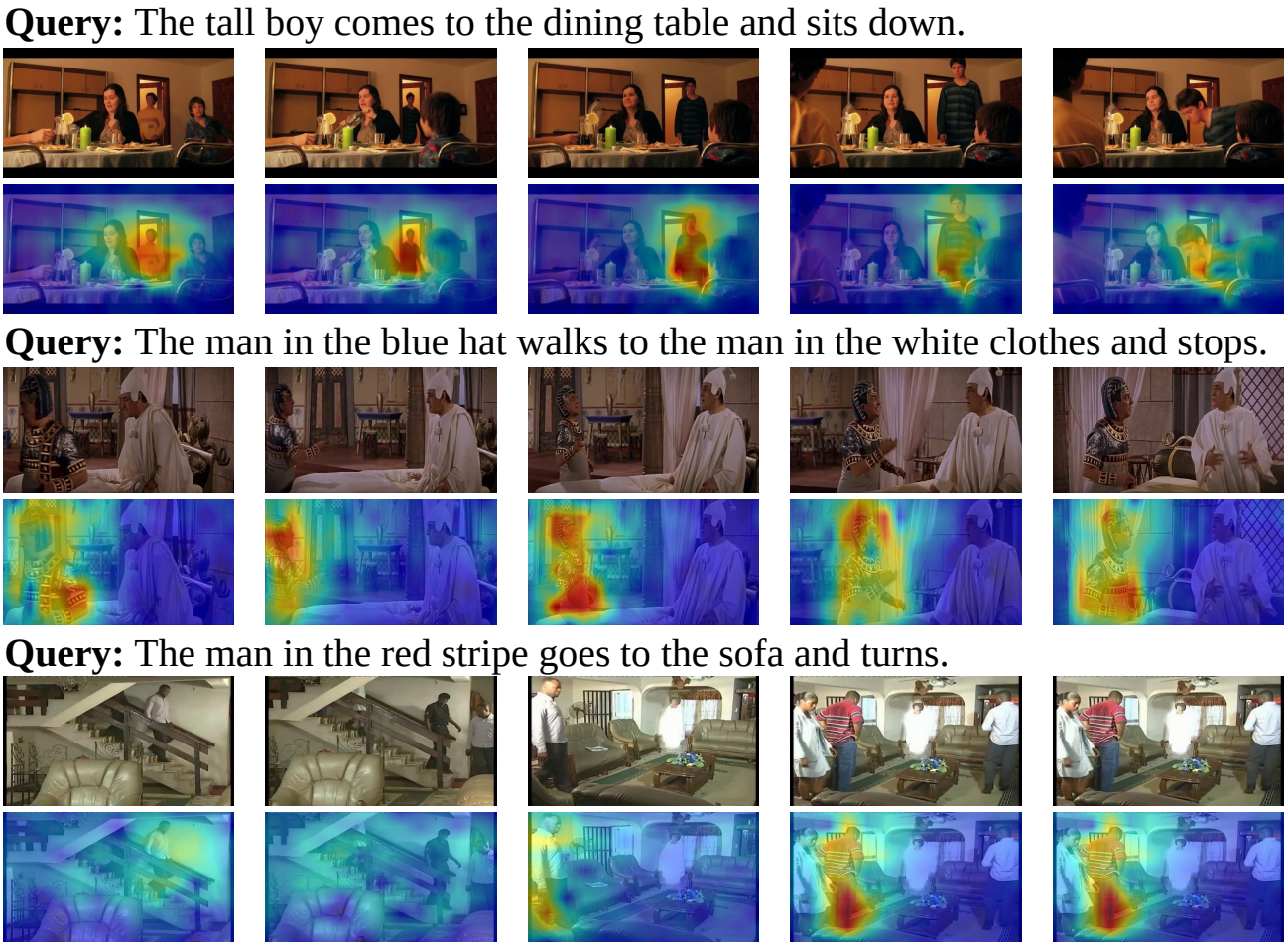

Figure S1. A visualization of the attention maps used in the static-to-dynamic information transmission block. For each case, we present the original frames and the visualized attention maps in the first row and the second row, respectively. The attention weights are normalized by the maximum attention values of all frames.

B. Further Implementation Details

B.1. Hyper-parameters

During training, we pre-extract video features with the video encoder (i.e., slowfast network [S2]) and save them to disk to improve training efficiency, thus we didn't finetune the video encoder during training. We set the learning rate as $1e^{-5}$ for the image encoder and the language encoder, and $1e^{-4}$ for the rest of the model. We train our model with AdamW optimizer [S5] with a weight decay of $1e^{-4}$. We follow MDETR [S4] to use exponential moving average (EMA) and set the decay weight as 0.999 for HCSTVG datasets [S7] and 0.99997 for VidSTG dataset [S11]. It cost about 2 days to train our model on 8 Nvidia RTX A6000 GPUs for VidSTG dataset using a batch size of 8. Different from previous state-of-the-art approaches TubeDETR [S8] and Augmented 2D-TAN [S6] which use complicated data augmentation strategies, we didn't use any data augmentation for simplicity. For the static stream, we uniformly sample T_s frames as input, and the inputted frames are resized to have a shorter side less or equal to N_{short} pixels and a longer side less or equal to $1.8 \cdot N_{short}$ pixels. We set $T_s = 48, N_{short} = 320$ for HCSTVG datasets [S7] and $T_s = 64, N_{short} = 448$ for VidSTG dataset [S11] (since it contains some long videos with small target objects). For the dynamic VL stream, we first pad the inputted frames to a square shape and then resize it to a resolution of 256×256. We follow the sampling strategy in 2D-TAN [S10] to sample the pre-extracted features to have a fixed temporal length T_d of 16 and 64 for HCSTVG [S7] and VidSTG datasets [S11], respectively. In the cross-stream collaboration blocks, we use bilinear interpolation to align the spatial resolution of feature maps outputted by different streams. For aligning the temporal resolution in the collab-

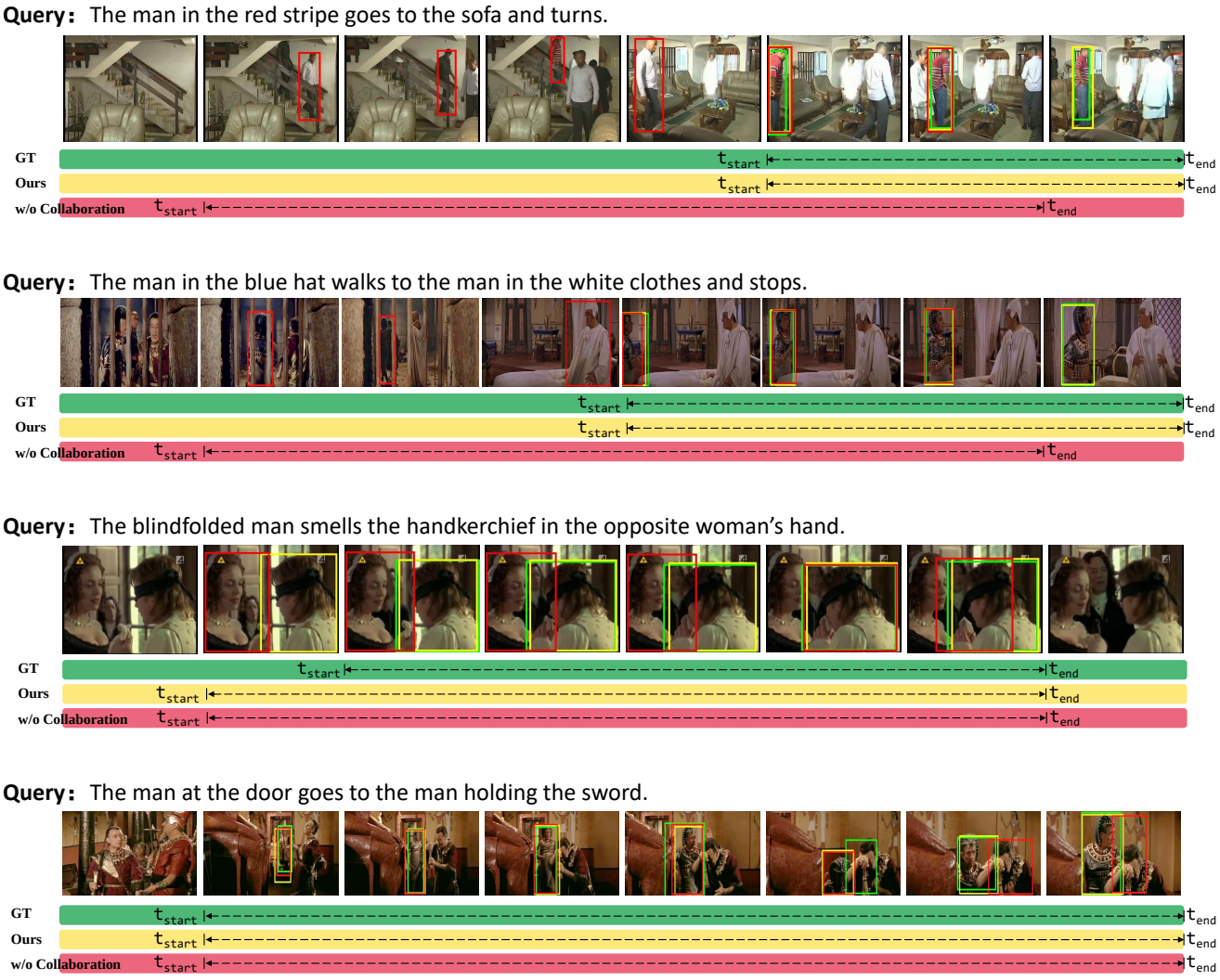

Figure S2. Visualization of the predicted tube of our approach (yellow), our approach without cross-stream collaboration block (red), and ground truth (green). In the first two samples, our baseline without collaboration block predicts the wrong temporal time span. In the last two samples, our baseline without collaboration block predicts wrong spatial bounding boxes. (Best viewed zoomed in on screen.)

oration blocks, we use temporal mean pooling and temporal replication for static-to-dynamic and dynamic-to-static information transmission blocks, respectively. In the static-to-dynamic information transmission block, in order to reduce the influence of using a too-sharp (caused by softmax) attention map A^i to guide the dynamic stream, we re-scale the query for calculating the attention maps by multiplying a factor α (it is similar to use a different temperature in softmax). α is set as 1.0 and 0.1 for VidSTG dataset [S11] and HCSTVG dataset [S7], respectively.

B.2. Detailed Formulation of L_{ta}

During training, we follow previous work [S9] to add a temporal attentive loss L_{ta} for accelerating the convergence. It encourages the model to predict a high matching score for those frames/clips inside the target temporal span. Specifically, we employ an FC on O_t^N to predict the matching score $\hat{S}_t^{frame}$ for the t-th sampled frame in the static stream. And we employ an MLP on $F_d[t]$ to predict the matching score $\hat{S}_t^{clip}$ for the t-th clip in the dynamic stream. Then L_{ta} is formulated as:

$$L_{ta} = \lambda_f L_{ta}^{frame} + \lambda_c L_{ta}^{clip}, \tag{1}$$

$$L_{ta}^{frame} = \frac{-\sum_{t=1}^{T_s} m_t^f \log \hat{S}_t^{frame}}{\sum_{t=1}^{T_s} m_t^f}, \tag{2}$$

$$L_{ta}^{clip} = \frac{-\sum_{t=1}^{T_d} m_t^c \log \hat{S}_t^{clip}}{\sum_{t=1}^{T_d} m_t^c}, \tag{3}$$

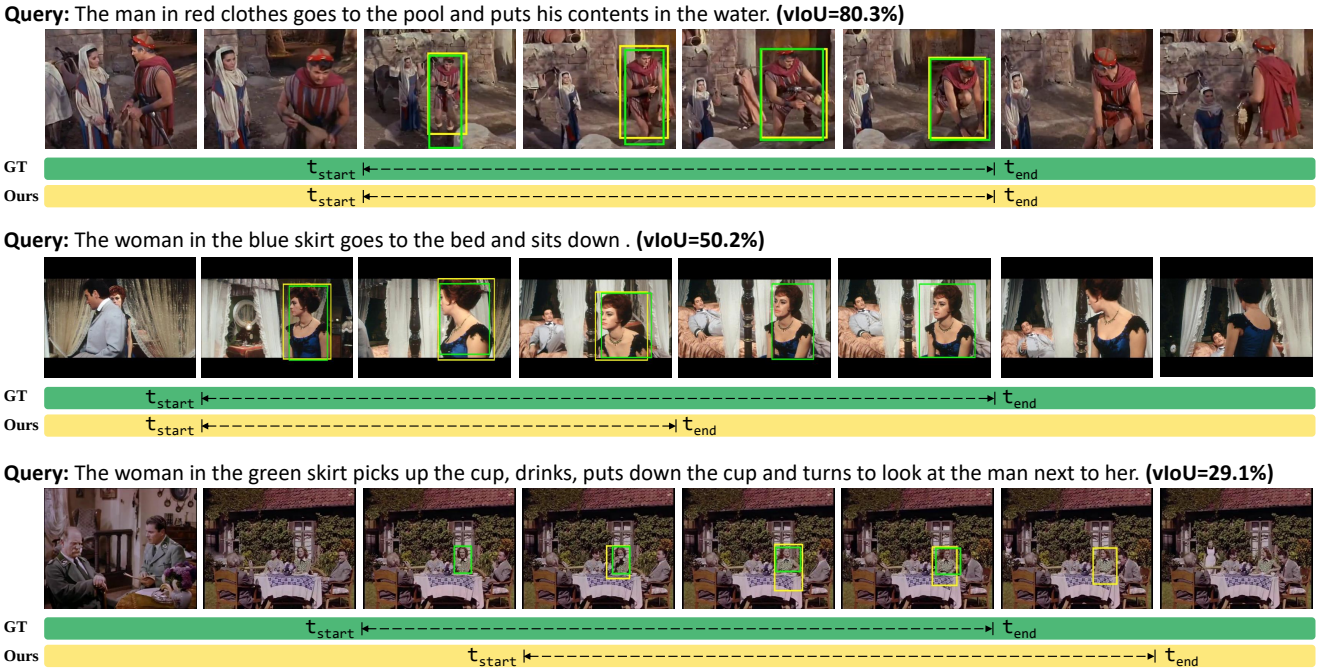

Figure S3. More visualization of our predicted results (yellow) and ground truth (green). (Best viewed zoomed in on screen.)

Table 1. Experiments on HCSTVG-v2 validation set using different pretrain versions of SlowFast.

Pretrain Dataset	m_vIoU	vIoU@0.3	vIoU@0.5	m_tIoU	m_sIoU
AVA [S3]	**38.7**	**65.5**	33.8	**58.1**	**65.7**
Kinetics-600 [S1]	38.4	64.7	**34.2**	57.6	**65.7**

Table 2. Experiments on HCSTVG-v2 validation set using different loss coefficients.

λ_3	λ_4	m_vIoU	vIoU@0.3	vIoU@0.5	m_tIoU	m_sIoU
5	0.1	37.6	63.3	33.1	56.6	**65.9**
5	1	**38.7**	65.5	33.8	**58.1**	65.7
5	10	37.1	62.8	29.5	57.2	64.2
1	1	38.0	64.3	32.6	57.1	65.7
20	1	38.6	**65.6**	**34.8**	57.9	65.7

where m_t^f, m_t^c are set as 1 if the t-th frame/clip is inside the target temporal time span, otherwise m_t^f, m_t^c are set as 0. λ_f, λ_c are weights for balancing the two losses, they are set as 0.5 and 1.0, respectively.

C. Additional Experiment Results

Effect of SlowFast pretraining. In our video encoder, we follow Aug. 2D-TAN [S6] to extract clip-level features using a SlowFast [S2] Network. It is pretrained on AVA dataset [S3] which is a dataset for the relevant spatio-temporal action localization task. We also conduct experiments on HCSTVG-v2 dataset using the Kinetics-600 [S1] pretrained SlowFast and results are presented in Table 1. As shown, the models pretrained on AVA dataset and Kinetics-600 achieve similar performances, indicating that the performance improvement of our model is not mainly from the pretraining on AVA dataset.

Effect of loss coefficients. We train our model with multiple losses and we balance the weights on $L_{l1}, L_{GIoU}, L_{tg}, L_{ta}$ with loss coefficients $\lambda_1, \lambda_2, \lambda_3, \lambda_4$, respectively. In our experiments, we keep λ_1, λ_2 consistents with previous works [S4, S8] and mainly tune λ_3, λ_4. The results are summarized in Table 2. We observe that the model achieves the best m_vIoU score with $\lambda_4 = 1$ and a value of λ_4 that is either too large or too small results in a significant drop in performance. Moreover, the model is less sensitive to changes in λ_3 within the range of $[1, 20]$.

Inference speed. It cost around 0.5s for inferencing a video around 20s with 5fps sampling rate using a Nvidia 3090 GPU. In our model, the required memory for processing a video grows linearly as the video duration increases. Thus, for inferencing very long videos (e.g. several hours), we have to split the video into some segments due to memory constraints, which is currently a limitation exists in both our approach and previous methods. In the future, we will consider developing spatio-temporal grounding methods that can efficiently localize the target object in long videos with low memory costs.

References

[S1] Joao Carreira, Eric Noland, Andras Banki-Horvath, Chloe Hillier, and Andrew Zisserman. A short note about kinetics-600. *arXiv preprint arXiv:1808.01340*, 2018. 4

[S2] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6202–6211, 2019. 2, 4

[S3] Chunhui Gu, Chen Sun, David A Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George Toderici, Susanna Ricco, Rahul Sukthankar, et al. Ava: A video dataset of spatio-temporally localized atomic visual actions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6047–6056, 2018. 4

[S4] Aishwarya Kamath, Mannat Singh, Yann LeCun, Gabriel Synnaeve, Ishan Misra, and Nicolas Carion. Mdetr-modulated detection for end-to-end multi-modal understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1780–1790, 2021. 2, 4

[S5] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 2

[S6] Chaolei Tan, Zihang Lin, Jian-Fang Hu, Xiang Li, and Wei-Shi Zheng. Augmented 2d-tan: A two-stage approach for human-centric spatio-temporal video grounding. *arXiv preprint arXiv:2106.10634*, 2021. 2, 4

[S7] Zongheng Tang, Yue Liao, Si Liu, Guanbin Li, Xiaojie Jin, Hongxu Jiang, Qian Yu, and Dong Xu. Human-centric spatio-temporal video grounding with visual transformers. *IEEE Transactions on Circuits and Systems for Video Technology*, pages 1–1, 2021. 1, 2, 3

[S8] Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid. Tubedetr: Spatio-temporal video grounding with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2, 4

[S9] Yitian Yuan, Tao Mei, and Wenwu Zhu. To find where you talk: Temporal sentence localization in video with attention based location regression. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9159–9166, 2019. 3

[S10] Songyang Zhang, Houwen Peng, Jianlong Fu, and Jiebo Luo. Learning 2d temporal adjacent networks for moment localization with natural language. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12870–12877, 2020. 2

[S11] Zhu Zhang, Zhou Zhao, Yang Zhao, Qi Wang, Huasheng Liu, and Lianli Gao. Where does it exist: Spatio-temporal video grounding for multi-form sentences. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2, 3