A Appendix

Here, we provide some more details about our approach along with additional results and visual analysis. We also include tables which we were not able to include in main paper due to space limitations.

- Section B: We address the challenges and limitations of detector and tracker.
- Section C: Qualitative Analysis on the model's predictions.
- Section D: We show more discussion and analysis.
- Section E: Training details about architectures, datasets, and, other hyperparameters.
- Section F: Qualitative Analysis on Detection and tracking, success and failure cases and analysis on the video in the wild.

B Challenges and Limitations

STVG datasets are extremely challenging, especially the HCSTVG-v1 and HCSTVG-v2 where even detection and tracking fails, shown by maximum upper bound achievable in Table 12b. The HCSTVG datasets contains sudden zoom shots, scene changes, and defocus, where even good detectors fail. The additional pre-processing to track the detections to generate tubelets introduce more noise and struggles to track the right person with person crossover, scene change (very high displacement in bbox leads it to assign different IDs), view change and only partial body availability. Due to these two main limitations, we propose to solve the task by breaking it into two sub-tasks. A future work involves exploiting temporal modeling associated with each individual object jointly; however, in our current approach, we show promising results quantitatively and qualitatively.

C Qualitative Analysis (Main architecture)

In Fig. 6, we show the effectiveness of our approach qualitatively. W-GDINO struggles with grounding the right actor as well as no temporal bounds, whereas our approach spatio-temporally grounds the actor better than baseline.

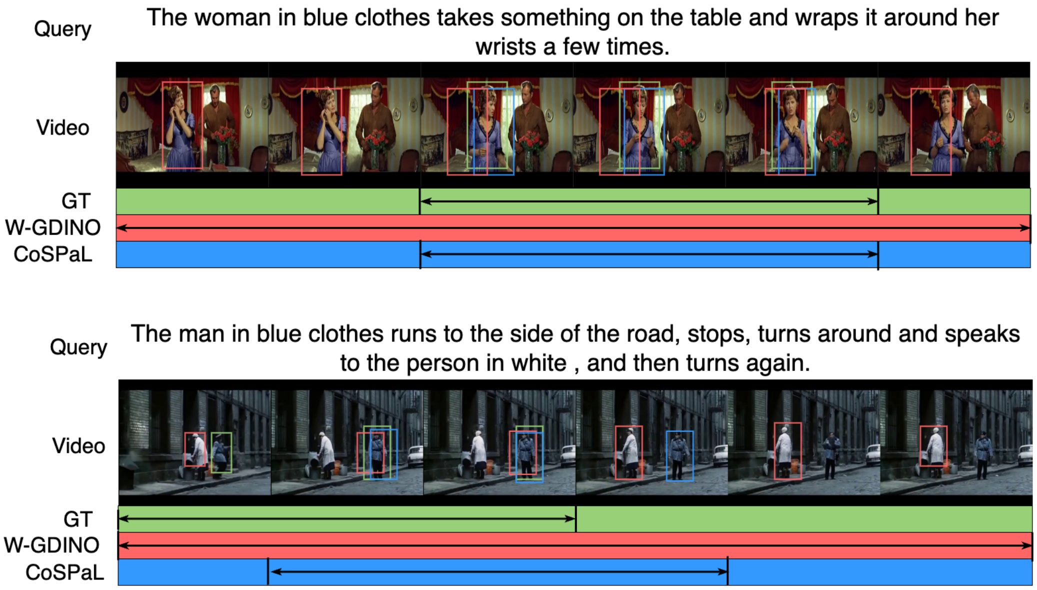
Query
The woman in blue clothes takes something on the table and wraps it around her wrists a few times.
Video
GT
W-GDINO
CoSPaL
Query
The man in blue clothes runs to the side of the road, stops, turns around and speaks to the person in white , and then turns again.
Video
GT
W-GDINO
CoSPaL

Figure 6: **Qualitative analysis:** We observe that W-GDINO detects without considering the context of the query, which is improved using the proposed method.

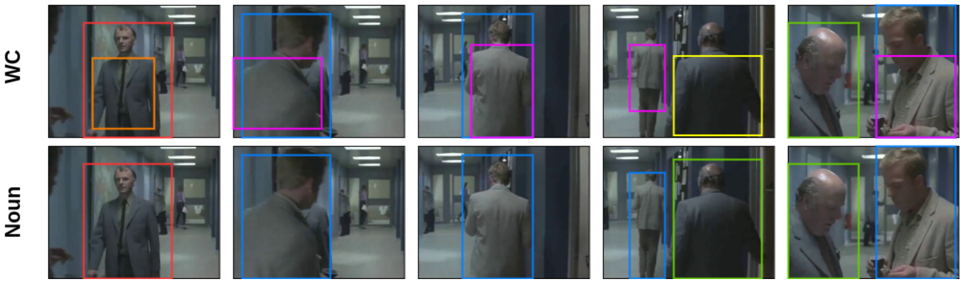

Figure 7: **Comparison between GDINO query: Whole caption (WC) vs Noun.** The first row shows detection boxes for whole query as input to the GDINO against noun extracted from the query in second row. We observe that it focuses on other objects (for eg. suit (shown in orange, pink, yellow)) which may not be the target instance but overlapping with target instance and thus helps in better score. (Tab 7). Query for the above video (WC): `The bald man leaves the room pulls the door walks towards the man in the white suit and then turns to face the white suit man.` Noun: `'man'.`

D Discussions

We include multiple discussions to support and strengthen the claims in our main paper:

Performance with Whole Caption (GDINO Input): In the main paper, we follow the traditional weakly supervised settings for *fair comparison* with previous SOTA, where at train and test time the detector outputs *ALL* human/object bounding boxes, and, given the query, the output should be object tubelet with maximum attention. In another setting, we analyze sending in the original caption and perform tracking on output detections. We have shown the difference in detection with only sending noun vs whole caption in Fig. 7. WC setting output detections which doesn't correspond to all subjects or overlapping detections to specific subject. In Table 7 we compare three settings. Training and testing on noun extracted from query (Noun), Train and test with whole caption (WC), and, finally, Train on WC and test with Noun. Looking at second row, input to Grounding DINO with extra information helps. To compare it with traditional weakly settings, third row we perform test with detections using Noun output. This study suggests that whole captions as query generates better detections Grounding DINO, although it might not adhere to traditional weakly-supervised settings.

Table 7: Grounding DINO Input: Noun vs Whole Caption.

Train	Test	tIoU	m_vIoU	vIoU@0.3	vIoU@0.5
Noun	Noun	37.6	19.2	28.8	15.3
WC	WC	34.5	22.7	32.5	18.2
WC	Noun	35.0	18.6	26.8	15.0

Improvement in performance with SPS: From Table 5, we consistently observe a 2-3% boost for each setting with inclusion of SPS. This shows that increasing scene understanding is complementary to both baseline and baseline+CRG settings. Going in-depth analysis, in Tables 8a - 8c, we show the improvement by SPS based training for all three settings - TPG only, CRG only, and, TPG + CRG. Self-paced learning boosts score in each of the settings by 2.4, 3.4, and, 5.0 respectively. This shows the efficacy how self-paced scene understanding training paradigm helps network become more discriminative with time both spatially and temporally. This is also corroborated by the fact that training via SPS paradigm outperforms single-stage training on the whole dataset (shared in Table 5 main paper).

Analysis on Text encoder: Grounding DINO finetunes the vision encoder but keeps the text encoder fixed. The vision backbone is fixed to Swin-T. For textual features, we explore two choices to find the best alignment between vision and text to begin with. From Table 9, BERT outperforms CLIP on the baseline settings, TPG. Thus, we choose BERT as encoder for all our experiments.

Table 8: Analysis on SPS in all three situations.

(a) TPG only.

Stages	m_tIoU	m_vIoU	v@0.3	v@0.5
I	34.1	17.7	26.0	14.4
II	36.2	18.5	27.0	14.8
III	38.2	20.1	28.5	17.6

(b) CRG only.

Stages	tIoU	m_vIoU	v@0.3	v@0.5
I	33.4	17.7	24.6	14.8
II	36.3	19.6	28.8	16.3
III	38.1	21.1	30.7	18.4

(c) TPS and CRG.

Stages	tIoU	m_vIoU	v@0.3	v@0.5
I	32.3	17.1	24.4	14.0
II	37.2	19.9	28.9	16.7
III	41.2	22.1	31.8	19.6

Table 9: Choice of Textual Encoder: CLIP vs BERT.

Encoder	tIoU	m_vIoU	vIoU@0.3	vIoU@0.5
CLIP	35.7	18.8	28.8	14.8
BERT	37.6	19.2	28.8	15.3

Study on Decoder Layer features: We perform an analysis on TPG with different decoder layer features. Since G-DINO shares architecture with DETR, we extract features from six layers of decoder and ran our baseline. In Table 10, we show the performance with features from different decoder layers. We observe features from decoder layer 1 performed the best. To further refine background noise, we restrict the number of tubelets for our settings to 10. The last row (Table 10) shows that it further boost the performance by 0.8%.

Standalone classification and temporal scores: We perform standalone analysis on classification accuracy and temporal grounding metrics from previous works (Zheng et al., 2022a;b; Lin et al., 2020) in Table 11. In classification accuracy, we observe our approach outperforms W-GDINO by 20% and baseline TPG by 3.2%. For temporal IoU metrics, we observe including contextual phrases boost the performance further at all IoUs.

Analysis on multiple IoUs: In Table 12a, we show performance comparison ranging from 0.1 till 0.7 on HCSTVG dataset. CoSPaL outperforms TPG and W-GDINO at all IoUs. Our proposed approach is more effective at higher IoUs, showing a gain of 4.3% and 4.1% at 0.5 and 0.7 IoU respectively. We perform similar analysis on VidSTG dataset comparing performance at multiple IoU ranging from 0.1 till 0.7. Tables 13a and 13b shows that proposed approach outperforms both W-GDINO and TPG at all IoUs.

Upper bound Analysis: To quantify how challenging HCSTVG-v1, HCSTVG-v2 and VidSTG datasets are, we perform an analysis to find the upper bound, that is maximum achievable results. This analysis is necessary since it tells how challenging detection and tracking is on these datasets. We set the temporal bound 100% from ground truth. Looking at Table 12b, if the network works perfectly, our proposed module can achieve max 62.3, 52.5, 45.3, 39.8 m_vIoU on HCSTVG-v1, HCSTVG-v2, VidSTG-Declarative, and, VidSTG-Interrogative respectively. With respect to that our current approach achieves effective performance of 35.4, 42.3, 28.5, 28.6 percentage of maximum achievable.

E Experiment Details

E.1 Detection and Tracking

Detector: Grounding DINO involves two hyperparameters namely text and box threshold. We set it to 0.4 for both. Setting a lower or higher values leads to oversampling or missed detections. Since dataset contains multiple resolution of images, we set the image width to 480 if original frame width is less than 550, else 800.

Tracker: The parameters set for BoTSORT tracker are: 1) new track threshold: 0.21, 2) Low track threshold: 0.1, 3) High track threshold: 0.34, 4) Matching threshold: 0.21, 5) Appearance threshold: 0.48, and, 6) Buffer frames: 60 to keep track of the object id for 60 number of frames.

Table 10: Comparison with different decoder layer features. Last row † shows further refinement to restrict upper bound on number of tubelets help.

Layer	m_tIoU	m_vIoU	vIoU@0.3	vIoU@0.5
I	**35.8**	**18.4**	26.7	**15.3**
II	35.4	18.0	**26.9**	15.0
III	35.6	17.7	25.7	14.3
IV	34.4	17.8	26.2	14.9
V	33.5	18.1	26.4	14.9
VI	34.6	17.9	26.1	15.2
I †	37.6	19.2	28.8	15.3

Table 11: Analysis on standalone classification accuracy and temporal IoU.

(a) Classification Accuracy.

Method	Acc.
W-GDINO	18.7
TPG	35.5
CoSPaL	38.7

(b) Temporal IoU.

TPG(Query)	NAV(Phrases)	IoU@0.1	IoU@0.3	IoU@0.5
✓		74.1	54.1	23.0
✓	✓	76.2	55.6	23.8

Table 12: Analysis on multiple factors showcasing effective of our proposed approach. In Table 12b, VidSTG-D means VidSTG Declarative and VidSTG-I means VidSTG Interrogative.

(a) Analysis on multiple IoUs on HCSTVG dataset.

Method	m_vIoU	v@0.1	v@0.2	v@0.3	v@0.5	v@0.7
W-GDINO	9.0	25.9	17.3	11.6	4.6	0.7
TPG	19.2	43.1	36.2	28.8	15.3	5.4
CoSPaL	22.1	45.6	38.7	31.6	19.6	9.5

(b) Upper-bound Analysis.

Dataset	m_tIoU	m_vIoU	vIoU@0.5
HCSTVG-v1	79.2	62.3	69.5
HCSTVG-v2	76.3	52.5	54.6
VidSTG-D	66.9	45.3	46.8
VidSTG-I	66.2	39.8	39.2

Table 13: Analysis on multiple IoUs showcasing effectiveness of our proposed approach.

(a) VidSTG-Declarative.

Method	m_vIoU	v@0.1	v@0.2	v@0.3	v@0.5	v@0.7
W-GDINO	10.6	25.0	17.6	13.0	7.8	4.1
TPG	12.9	28.2	20.9	16.2	9.9	5.6
CoSPaL	16.0	33.6	25.8	20.1	13.1	7.8

(b) VidSTG-Interrogative.

Method	m_vIoU	v@0.1	v@0.2	v@0.3	v@0.5	v@0.7
W-GDINO	9.8	23.2	16.5	12.2	6.7	3.5
TPG	11.4	26.8	18.8	14.0	8.0	4.5
CoSPaL	13.5	30.3	22.0	16.4	10.2	5.7

E.2 Architecture Hyperparams settings

Weakly-GDINO: For weakly-GDINO, we input whole text as the query and frame from video as image input. Frames are sample with a stride of 5. To calculate the GDINO predictions for a video, Firstly, we run the tracker to generate all tubelets in the video. To evaluate, we average the confidence of each tubelet across temporal dimension. The predicted tubelet is assigned to the the tubelet with highest average confidence score. The starting and ending timestamp of the predicted tubelet is used for temporal IoU calculation.

Tubelet Phrase Grounding: It contains two modules - spatial and temporal grounding. The batch size is set to 32. In spatial grounding module, we use Adam optimizer with a learning rate of 1e-4. The maximum length for number of words in text is set to 25 for HCSTVG. Temporal grounding module had Adam optimizer with learning rate 4e-4.

Contextual Referral Grounding We use GPT-3.5 to extract referral tubelet attributes (Q_{oa}) and referral tubelet action verbs (Q_{ov}). The input query Q_a and Q_v to the GPT to extract Q_{oa} and Q_{ov} respectively as below:

Q_a: Extract the quantifier phrase describing the main person.
Q_v: Break the complex actions into simpler actions.

We provide few examples of original texts and extraction from GPT-3 for both scenarios. For first case, extraction of main obejct in context and attributes related to its are as follows:

$Q1$: The bearded woman walks to the woman in gray clothes and touches her face.
$A1$: The bearded women.
$Q2$: The man in the brown hat drops the hat of the man in the black hat then pushes the opposite man then turns and punches the man in the back.
$A2$: The man in the brown hat.
$Q3$: The woman with yellow hair walks from the right to the left of the man in leather then pulls his arm away.
$A3$: The woman with yellow hair.

In case of main actor and it's attribute extraction, GPT-3 worked perfectly. However, breaking complex actions into sub-actions, GPT-3 faced challenges and sometimes hallucinates which activity belongs to which actor. One ***success*** case as follows:

$Q1$: The bald man leaves the room pulls the door walks towards the man in the white suit and then turns to face the white suit man.
$P1$: The bald man leaves the room.
$P2$: He walks towards the man in white suit.
$P3$: He turns to face the white suit man.

One ***failure*** case as follows:

$Q1$: The man in the black military uniform catches the things thrown by the opposite man with both hands turns and bends over to pick up his hat and puts on it.
$P1$: The man in the black military uniform catches the things.
$P2$: He throws the thing.
$P3$: He turns and bends over.
$P4$: He pick up his hat.

In above scenario, P2 relates to the activity by the actor not in main context. We filter out these phrases by looking into verbs in active tense. Those verbs denote activity performed by the main actor.

Self-paced Scene understanding: In SPS curriculum based learning, we set the upper bound on the number of object tubelets per video. The first stage bound is set to videos with only upto 4 tubelets and it's incremented by 3 in each stage for two more stages. In last stage, the number of tubelets is 10 and it contains all the videos.

E.3 Compute requirements

For our work, we run our models on single 16 GB Tesla V100 GPU with a batch size of 32. The training time for HCSTVG-v1 is 4-5 hours, HCSTVG-v2 id 7-8 hours and VidSTG it's 10-12 hours.

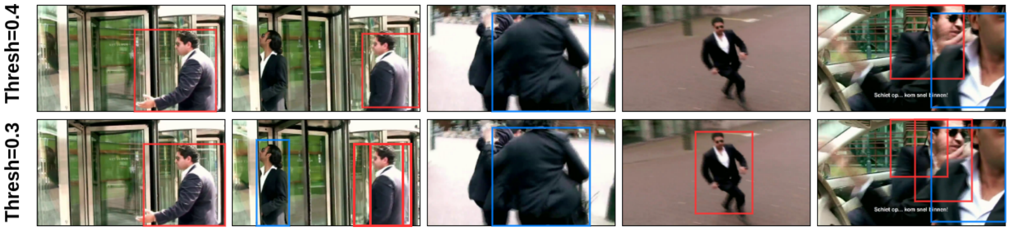

Figure 8: **Comparison between threshold for GDINO:** The first row shows detection boxes with threshold set to 0.4 and the second row shows the detection with threshold set to 0.3. We see few missed detections in earlier case, however, in later, overlapping detection issues arises. Even in second scenario, in third frame lowering confidence didn't help. The detection was missed. Query text: Noun: `'man'`.

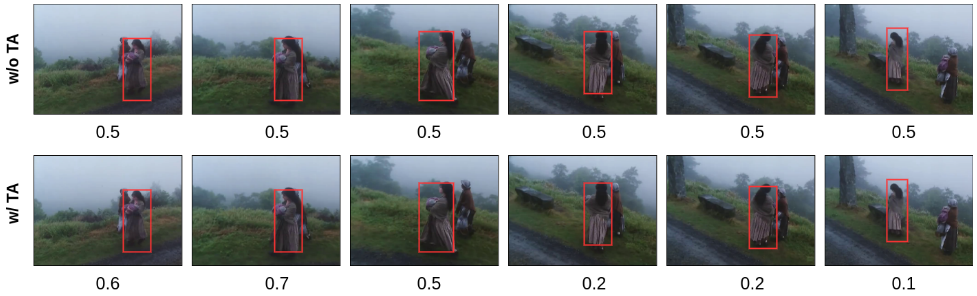

Figure 9: **Effect of Temporal Attention:** Without temporal attention (w/o TA) in first row, we observe that each frame gets equal weight, however, utilizing temporal attention (w/ TA, second row) increases weight on key frames and decrease weight for non-important frames in relation to query. `Query: The` **`woman holding the child`** `walks to the side of a stone bench stops hands the child to the woman next to her and walks to the front of the stone bench`

E.4 Societal Impact

The proposed work could be used for surveillance and if the query is not descriptive enough can ground the wrong person leading to possible harm. However, on the positive aspect, the proposed work is free of biasness issues due to use of foundation models (trained on bigger datasets) and can be deployed in wild.

F Qualitative Analysis

F.1 Failures in Detection and tracking

In this qualitative analysis, we show the inherent failure of Grounding DINO(Liu et al., 2023) and tracker (Aharon et al., 2022).

F.1.1 Detection Failure

In Fig. 8 we show that GDINO fails to detect the person. If we reduce threshold, it is able to detect, but, then it leads to overlapping detections which will add one another step of post-processing of non-maxima suppression.

F.1.2 Tracking Failure

There are two type of failure that happens in tracking: 1) Assigning same ID to different objects, and, 2) Different IDs to same objects. In both scenarios, tubelet features get impacted. Fig. 13 illustrates both the failures.

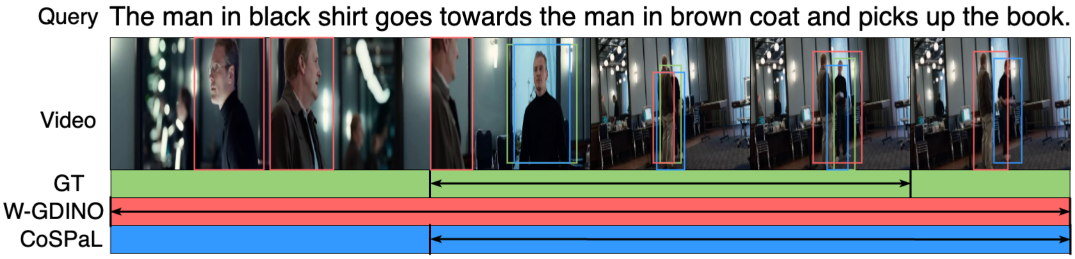
Query The man in black shirt goes towards the man in brown coat and picks up the book.
Video
GT
W-GDINO
CoSPaL

Figure 10: **Qualitative Analysis:** W-GDINO struggles to attend to the query and switch between actors across time. Our proposed approach is able to detect the main actor in context (from textual query) almost correctly spatio-temporally.

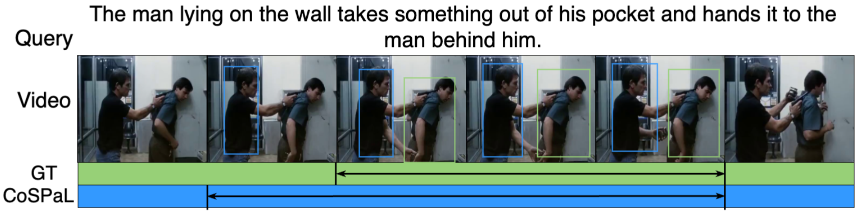
Query The man lying on the wall takes something out of his pocket and hands it to the man behind him.
Video
GT
CoSPaL

Figure 11: **Qualitative Analysis (*Failure scenario*):** In these scenarios, visual features are quite similar and query description is challenging to extract the attributes related to the main actor in context.

F.2 Effect of Temporal Attention

In this analysis we show how temporal attention applied over tubelet helps. Fig 9 shows impact of with and without temporal attention. With temporal attention across temporal dimension, key frames that has higher mutual information in relation to query is given higher weight.

F.3 Random Video Analysis - In the Wild

We take a random video from the internet and run our proposed approach. In Fig. 10, we show the comparison between ours against W-GDINO. We pick a video from a movie scene Steve Jobs and ran our detector and tracker and then use trained weights to predict the tubelet given the query. We formulate the query and video length on our own for this experiment.

F.4 Success and Failure cases

Fig. 11 shows a failure scenario of our model. We observe model fails when query description doesn't explicitly contains specific attributes describing the main actor in context and spatial features of objects are very similar.

Fig. 12 shows a success scenario. In first example (*top row*), since the model doesn't contains any information about background or other actors, W-GDINO in this scenario works. However, since it doesn't have understanding of time, our approach is temporally localize the action. *Bottom row* shows a challenging example where our method performs better. In general, proposed approach works good when the query contains attributes related to main actor (referral). This shows that our proposed use of Contextual Referral grounding aspect helps in the scenario.

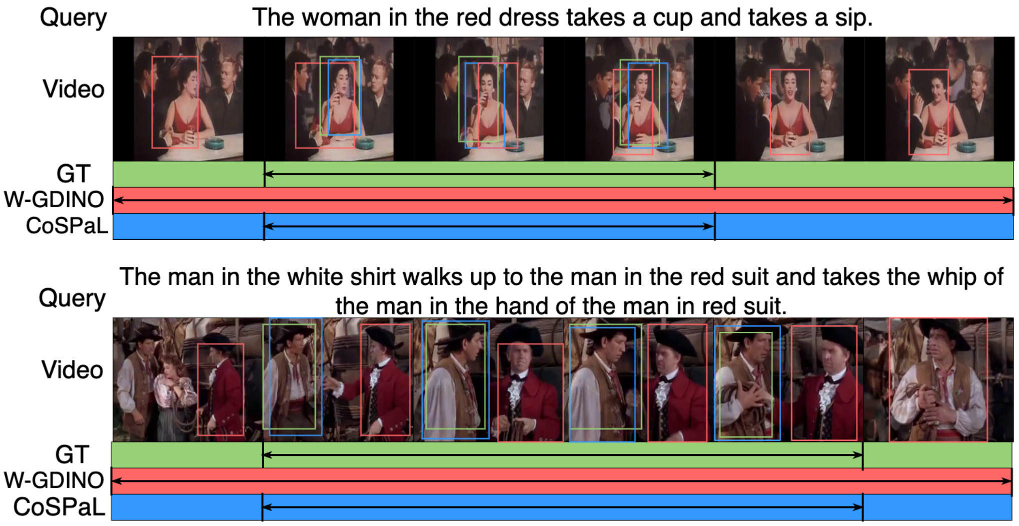
Query
The woman in the red dress takes a cup and takes a sip.
Video
GT
W-GDINO
CoSPaL
Query
The man in the white shirt walks up to the man in the red suit and takes the whip of the man in the hand of the man in red suit.
Video
GT
W-GDINO
CoSPaL

Figure 12: **Qualitative Analysis (*Success scenario*):** The proposed approach is able to properly spatio-temporally localize the actor and activity associated with it. *Top Row:* shows an easy example where W-GDINO also succeeds since the query contains description about one actor. However, it lacks temporal understanding and thus unable to localize the activity temporally. *Bottom row:* It shows a very hard example where there are query contains description about multiple actors in context. W-GDINO focuses on the background actor whereas our work is able to properly spatio-temporally localize the correct tubelet (referral tubelet).

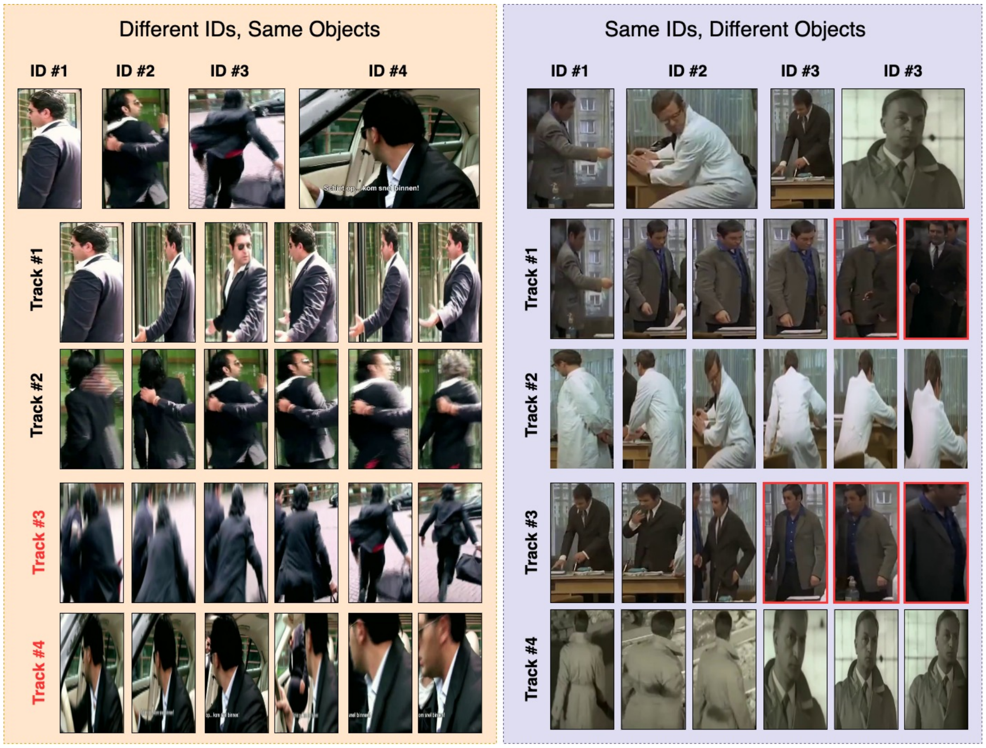

Figure 13: **Tracking failures:** *Left:* Different IDs, Same Objects - Tracks in red color are repetition of same earlier ID but assigned a new track. Tracks 1 and 4 are same IDs, and, tracks 2 and 3 are same IDs, but assigned different track IDs; *Right:* Same IDs, Different Objects - red boxes denotes switching of ID happened. Same id is assigned even if the object/actor is different.