Losses. During TubeR training, we first produce an optimal bipartite matching δ between predictions and ground truth tubelets. δ(i) is the index of the prediction matched with the i-th ground-truth tubelet. We need to calculate the losses between a set of ground-truth tubelets \( Y = (y_{\text{coord}}, y_{\text{switch}}, y_{\text{class}}) \) and the matched predictions \( \hat{y} = (\hat{y}_{\text{coord}}, \hat{y}_{\text{switch}}, \hat{y}_{\text{class}}) \).

We utilize four losses: an action classification loss, a box matching loss, a generalized IoU [3] loss and an action switch loss to train TubeR. The total loss is a linear combination of the four losses:

\[
\mathcal{L} = \lambda_1 \mathcal{L}_{\text{switch}}(y_{\text{switch}}, \hat{y}_{\text{switch}}) + \lambda_2 \mathcal{L}_{\text{class}}(y_{\text{class}}, \hat{y}_{\text{class}}) + \lambda_3 \mathcal{L}_{\text{box}}(y_{\text{coord}}, \hat{y}_{\text{coord}}) + \lambda_4 \mathcal{L}_{\text{iou}}(y_{\text{coord}}, \hat{y}_{\text{coord}}).
\]

(1)

\[
\mathcal{L}_{\text{class}} = - \sum_{i=1}^{N} \sum_{j=1}^{L} \{y_{\text{class}}(i, j) \log \hat{y}_{\text{class}}(\delta(i), j) + (1 - y_{\text{class}}(i, j)) \log (1 - \hat{y}_{\text{class}}(\delta(i), j)) \}.
\]

(2)

\[
\mathcal{L}_{\text{switch}} = - \sum_{i=1}^{N} \sum_{j=1}^{T_{\text{out}}} \{y_{\text{switch}}(i, j) \log \hat{y}_{\text{switch}}(\delta(i), j) + (1 - y_{\text{switch}}(i, j)) \log (1 - \hat{y}_{\text{switch}}(\delta(i), j)) \}.
\]

(3)

\[
\mathcal{L}_{\text{box}} = \sum_{i=1}^{N} \sum_{j=1}^{T_{\text{out}}} \| y_{\text{coord}}(i, j) - \hat{y}_{\text{coord}}(\delta(i), j) \|_1.
\]

(4)

\[
\mathcal{L}_{\text{iou}} = \sum_{i=1}^{N} \sum_{j=1}^{T_{\text{out}}} \Phi_{\text{iou}}(y_{\text{coord}}(i, j), \hat{y}_{\text{coord}}(\delta(i), j)).
\]

(5)

\[
\Phi_{\text{iou}}(b, \hat{b}) = 1 - \frac{|b \cap \hat{b}|}{|b \cup \hat{b}|} - \frac{|B(b, \hat{b})|}{B(b, \hat{b})}.
\]

(6)

Here \( \Phi_{\text{iou}}(b, \hat{b}) \) is the generalized IoU [3] loss between two given boxes \( b \) and \( \hat{b} \). We empirically set the scale parameter as \( \lambda_1 = 1, \lambda_2 = 5, \lambda_3 = 2, \lambda_4 = 2 \).

*Equally contributed.
long-term context without tubelets. Tubelet design not only brings performance gain, but also directly predicts tubelets without an offline linker. Our long-term context features are effective for long videos with shot changes. It results in a modest parameter increase from 70.1M to 84.3M, which is lower than most two-stage models.

Visualization. We show more action tubelets generated by TubeR in Figure 2. TubeR performs well in various cases. In Figure 2 (a-b), we show the cases with deformable actors and crowded people from UCF101-24. Figure 2 (c-d) present the fast action and interacted action from JHMDB51-21. Moreover, some challenging cases on AVA are visualized in Figure 3. All these cases show our TubeR is able to generate precise tubelets with various length.

References

[1] Chunhui Gu, Chen Sun, David A Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George
Figure 3. Action tubelets visualization on AVA. We use different colors to mark different tubelets. Each action tubelet contains its action labels and boxes on each frame. We only show the action labels on the first frame of an action tube. We show some challenging cases here. (a) and (b) Raw actions: “play musical instrument”, “hug (a person)”. (c) Tiny actions. The actors are very tiny. (d) Crowded cases. (e-h) Shot cuts. All these cases show our TubeR is able to generate precise tubelets with various length.