Supplemental Material - NUTA: Non-uniform Temporal Aggregation for Action Recognition

Xinyu Li* Chunhui Liu* Bing Shuai Yi Zhu Hao Chen Joseph Tighe
{xxnl, chunhuil,bshuai,yzaws,hxen,jtighe}@amazon.com
Amazon Web Services

Different I3D Backbone. We give the definition of I3D-50 with temporal down-sampling (Table a) and I3D-50 without temporal down-sampling (Table b).

Temporal Modeling. To understand how the proposed NUTA unit works to aggregate temporal features, we visualize the temporal projection matrix. when the input clip has smooth transition, which indicates the information is roughly uniformly distributed over time, the NUTA unit performs uniform sampling (Figure 1). Most previous works based on 3D convolutions also perform temporal information aggregation in this way. When the input features have scene changes or contain noises (e.g. transition frames), the temporal mapping generated by NUTA stays to focus on representative information by skipping noises (Figure 2). When the frames are highly repetitive or contain only background without useful information for action recognition, the NUTA unit is able to skip the non-informative frames and focus more on the representative features (Figure 3).

Stage	Details	Output Size
Conv_1	$5 \times 7 \times 7$, 64, stride 2, 2, 2	$16 \times 112 \times 112 \times 64$
Maxpool_1	$2 \times 3 \times 3$, stride 1, 2, 2	$8 \times 56 \times 56 \times 64$
3*Res_2	$3*\begin{pmatrix} 3 \times 1 \times 1, 64 \\ 1 \times 3 \times 3, 64 \\ 1 \times 1 \times 1, 256 \end{pmatrix} \times 3$	3*$8 \times 56 \times 56 \times 256$
Maxpool_2	$2 \times 3 \times 3$, stride 2, 2, 2	$4 \times 28 \times 28 \times 256$
3*Res_3	$3*\begin{pmatrix} 3 \times 1 \times 1, 128 \\ 1 \times 3 \times 3, 128 \\ 1 \times 1 \times 1, 512 \end{pmatrix} \times 4$	3*$4 \times 28 \times 28 \times 512$
3*Res_4	$3*\begin{pmatrix} 3 \times 1 \times 1, 256 \\ 1 \times 3 \times 3, 256 \\ 1 \times 1 \times 1, 1024 \end{pmatrix} \times 6$	3*$4 \times 14 \times 14 \times 1024$
3*Res_5	$3*\begin{pmatrix} 3 \times 1 \times 1, 512 \\ 1 \times 3 \times 3, 512 \\ 1 \times 1 \times 1, 2048 \end{pmatrix} \times 3$	3*$4 \times 7 \times 7 \times 2048$
Average Pool	$8 \times 7 \times 7$	2048
FC	2048×400	400

(a) The detail structure and parameters of the I3D-50 network with temporal down-sampling that we used.

Stage	Details	Output Size
Conv_1	$5 \times 7 \times 7$, 64, stride 1, 2, 2	$32 \times 112 \times 112 \times 64$
Maxpool_1	$1 \times 3 \times 3$, stride 1, 2, 2	$32 \times 56 \times 56 \times 64$
3*Res_2	$3*\begin{pmatrix} 3 \times 1 \times 1, 64 \\ 1 \times 3 \times 3, 64 \\ 1 \times 1 \times 1, 256 \end{pmatrix} \times 3$	3*$32 \times 56 \times 56 \times 256$
3*Res_3	$3*\begin{pmatrix} 3 \times 1 \times 1, 128 \\ 1 \times 3 \times 3, 128 \\ 1 \times 1 \times 1, 512 \end{pmatrix} \times 4$	3*$32 \times 28 \times 28 \times 512$
3*Res_4	$3*\begin{pmatrix} 3 \times 1 \times 1, 256 \\ 1 \times 3 \times 3, 256 \\ 1 \times 1 \times 1, 1024 \end{pmatrix} \times 6$	3*$32 \times 14 \times 14 \times 1024$
3*Res_5	$3*\begin{pmatrix} 3 \times 1 \times 1, 512 \\ 1 \times 3 \times 3, 512 \\ 1 \times 1 \times 1, 2048 \end{pmatrix} \times 3$	3*$32 \times 7 \times 7 \times 2048$
Average Pool	$8 \times 7 \times 7$	2048
FC	2048×400	400

(b) The detail structure and parameters of the I3D-50 network without temporal down-sampling that we used.

*Equally contributed.

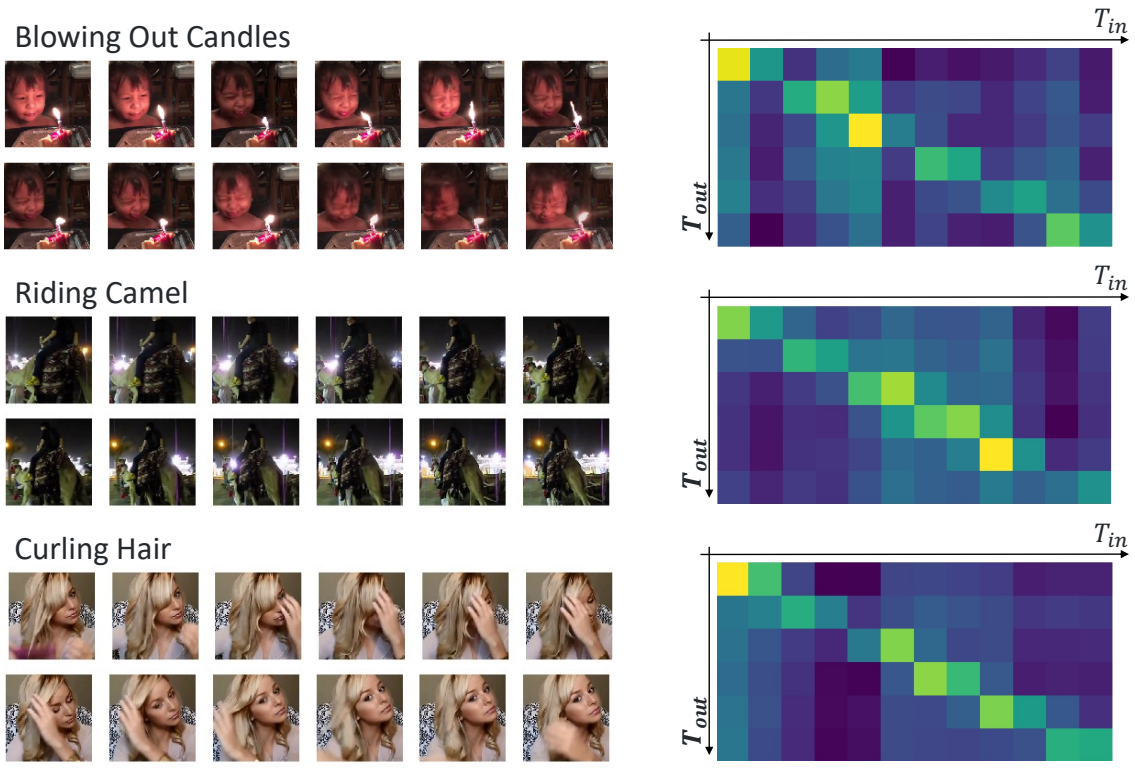

Figure 1: Temporal Modeling Example 1/3. In common continuous scenes, the first self-attention model will sample the frames in a roughly uniform manner, so that original motion features will be preserved.

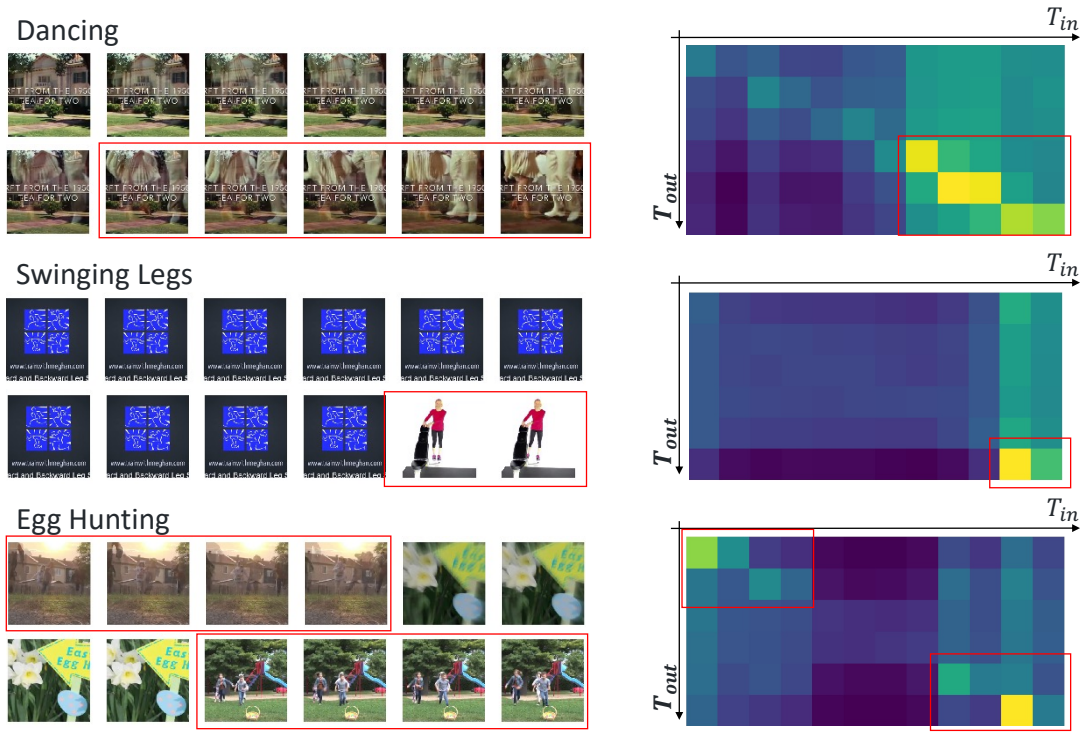

Figure 2: Temporal Modeling Example 2/3: Here we demonstrate our model output in the presence of shot boundaries. In the first two examples, Our model is able to associate each scene to its own temporal sub-clip. In the third example, our network focuses on the first and last shots and ignores the middle. This behaviour can not be learnt by 3D convolution kernels.

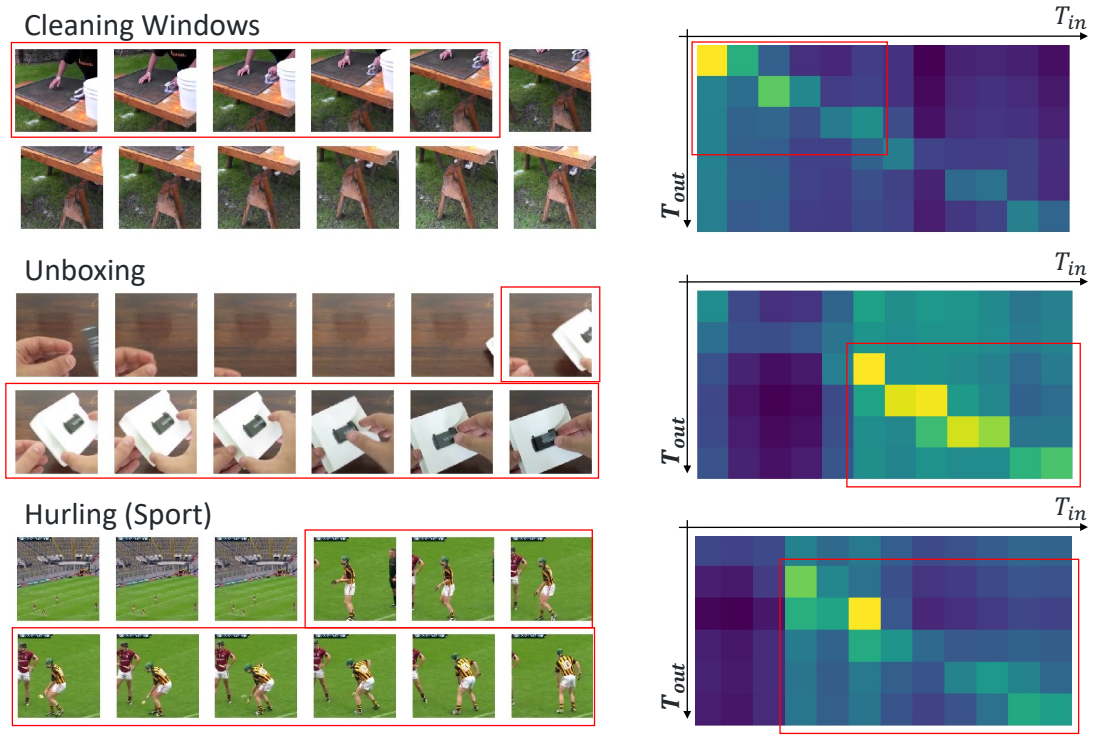

Figure 3: Temporal Modeling Example 3/3: When there are obviously irrelevant frames, our self-attention model can learn to skip these, and focus on the later frames that contain only relevant action information. This cannot be learnt if we just slide 3D kernels uniformly in a traditional 3D CNN way.