An Image is Worth 16x16 Words, What is a Video Worth?

Gilad Sharir, Asaf Noy, Lihi Zelnik-Manor

This paper proposes a simple adaptation of image transformers for video understanding. The proposed method matches the training and inference conditions, and no matter the size of the input video, they only uniformly sample 16 frames from the whole duration. Each input frame is then divided into 16 x 16 patches, each patch is embedded using a linear layer, with a learned positional embedding that is added to maintain to the ordering of the patches. After adding the CLS token that will play the role of a representation of each frame, each patched & embedded input image is then passed through a spatial transformer, which is the same as the transformer used for image recognition. The new addition is a temporal transformer added on top of the outputs of the spatial transformer applied to each frame independently. The inputs of the temporal are the CLS tokens of each one, with an added a video level CLS token, so inputs of shape D x (T+1). The temporal transformer temporally mixes the inputs and is followed by a classification head to produce the predictions.

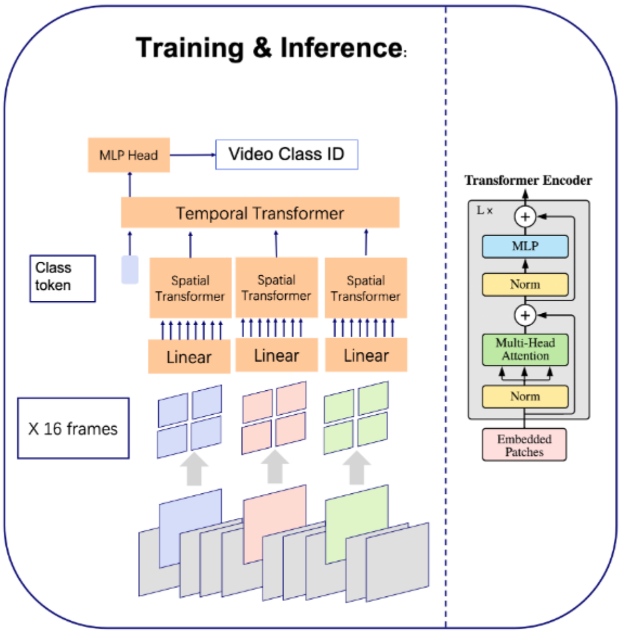

Figure 3. **Our proposed Transformer Network for video**

This model is quite efficient since the added temporal transformer operates over a smaller input sequence compared to that of the spatial transformer (for example 16x16=256 patches compared to 16 frames). Additionally, the number of time layers is half that of the spatial ones (12 vs 6).

VTN (Video Transformer Network)

Daniel Neimark, Omri Bar, Maya Zohar, Dotan Asselmann

VTN is very similar to STAM (Space-Time Attention Model) with a two-stage approach and starts by applying the spatial encoder first followed by a temporal transformer. However, in the temporal transformer, they do not use the standard quadratic attention mechanism, they use LongFormer layers with sliding window attention (only fetching information from the neighborhood) and a global one where some pre-selected tokens get access to all of the rest tokens, and such tokens are task-dependent. For classification, for example, the CLS token gets access to all of the rest since it is the one used at the end. Since both of the window sizes, the number of pre-fixed tokens is very small, the complexity is linear instead of quadratic.