Context-Aware Skeleton-based Action Recognition via Spatial and Temporal Transformer Networks

CS231N - Spring 2021

Sharan Ramjee, Sofian Zalouk

Human Action Recognition

- Motivation:
  - Human action recognition is an important task in video understanding
  - Various applications: Robotics, smart homes, autonomous driving, healthcare monitoring, augmented reality, security and surveillance, etc.

- Classification task:
  - **Inputs**: RGB(+D) video
  - **Outputs**: Action class

"Handshake"

Technical Challenges

- Variable size inputs:
    - Need to deal with arbitrarily long video sequences
- No perfect representation of data:
    - Need to model dynamics of important positions over time
- Setups are not always consistent:
    - Need to deal with variations in lighting, subjects, views, etc.
- Subtle differences among actions:
    - Need to classify similar actions (standing up vs sitting down)
- Action sequences can be long and contain multiple different actions:
    - Need to be able to attend to various parts of input

Skeleton-based Action Recognition

- Pose Estimation models:
    - **Input**: RGB(+D) video data
    - **Output**: Pose skeletons for each frame
- Skeletons are essentially graphs:
    - Much more natural representation of video data
    - Allow modeling the dynamics of joint positions over time
    - Better discriminative capabilities of models trained with graphs
    - Significantly reduces the dimensionality of video input data (faster)
- Graph Convolutional Networks (GCNs) + Recurrent Neural Networks (RNNs):
    - GCNs produce rich spatial features from skeletons
    - RNNs model long and short term relationships of skeletons over time
    - Spatial and temporal attention facilitate focusing on important features

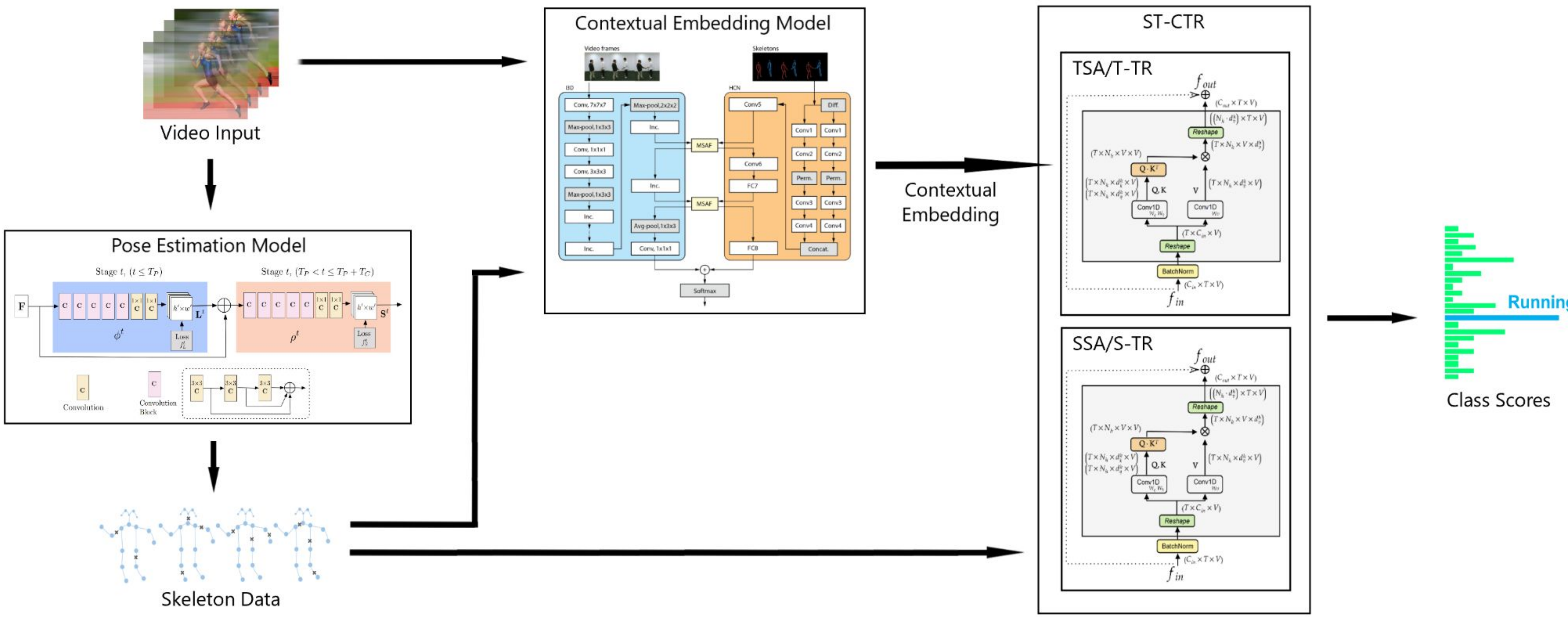
Related Works

- Global Context-aware Attention LSTMs (GCA-LSTMs) [1]:
  - Use contextual embeddings from video data that is fed into classifier
  - **Limitation**: Do not use GCNs; poor spatial features, very slow inference
- Spatial-Temporal Graph Convolutional Networks (ST-GCNs) [2]:
  - Connect corresponding joints across frames
  - Use GCNs to obtain features that are fed into a classifier
  - **Limitation**: Do not use RNNs; cannot predict skeletons for future frames
- Spatial-Temporal Transformer Networks (ST-TRs) [3]:
  - Use transformer self-attention to model both spatial and temporal dependencies between joints
  - **Limitation**: Do not leverage video context; leads to poor generalization

Spatial-Temporal Context-aware Transformer Network (ST-CTR)

- Pose Estimation Model (OpenPose) **[NOT USED]**:
  - Video data → Skeleton data
- Contextual Embedding Model (MSAF):
  - Video data + Skeleton data → Contextual embeddings
- Spatial Transformer (S-TR):
  - Skeleton data → Spatial features
- Temporal Transformer (T-TR):
  - Skeleton data + Contextual embeddings → Temporal features
- Softmax classifier:
  - Spatial features + Temporal features → Action classes

Spatial-Temporal Context-aware Transformer Network (ST-CTR)



Contextual Embedding Model

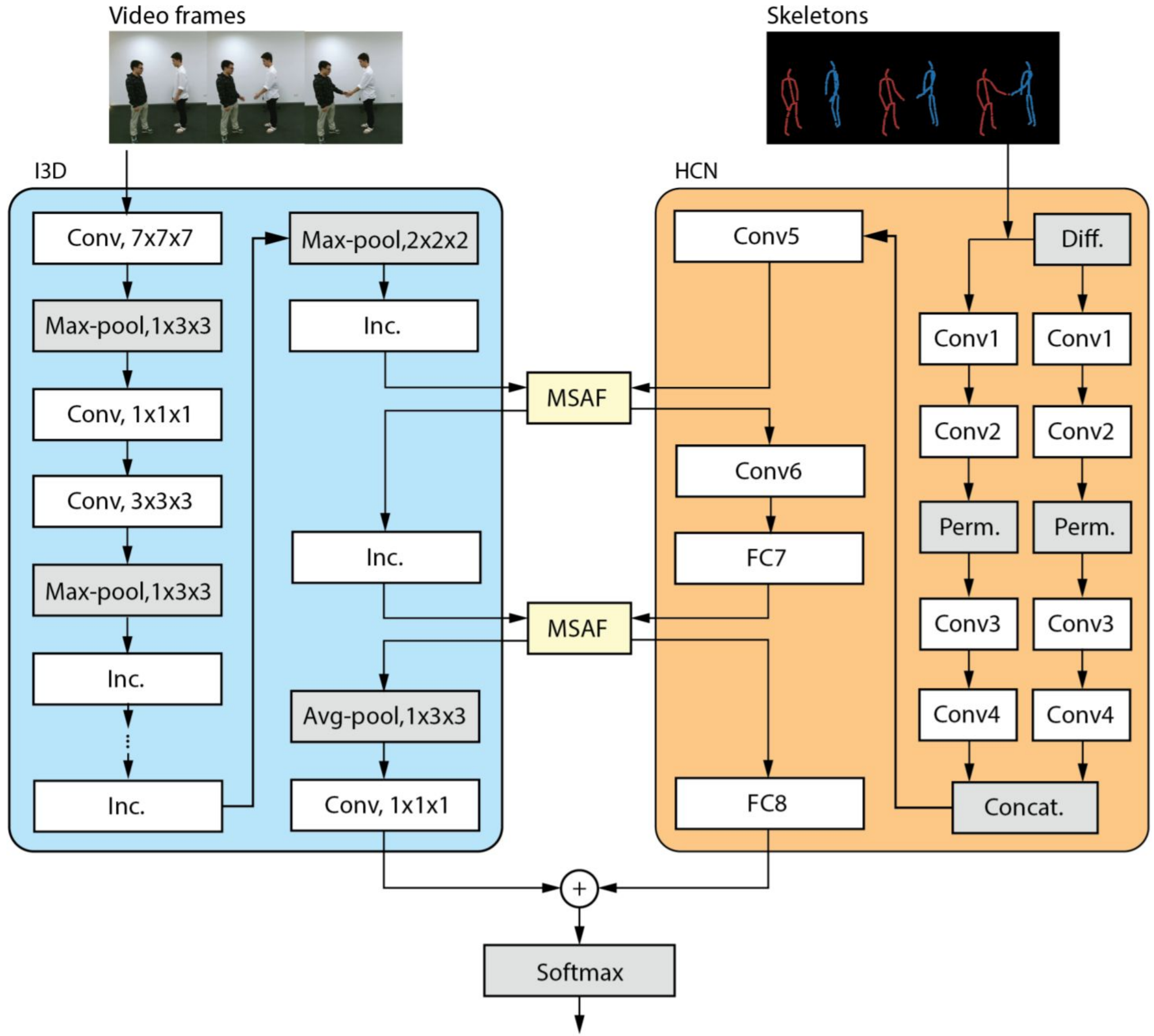
- Multimodal Split Attention Fusion (MSAF) [4]:
  - Splits the video/skeleton modalities into channel-wise equal feature blocks
  - Generates a joint representation that is used to generate soft attention for each channel across the feature blocks

- Modalities:
  - **Video stream**: I3D model [5]
  - **Skeleton stream**: HCN model [6]

- Two MSAF modules deployed:
  - **Intermediate level**: Early fusion style with 64 channels per block
  - **High level**: Late fusion style with 256 channels per block

- Hyperparameters:
  - **Suppression power (λ)**: 0.5

Multimodal Split Attention Fusion (MSAF)



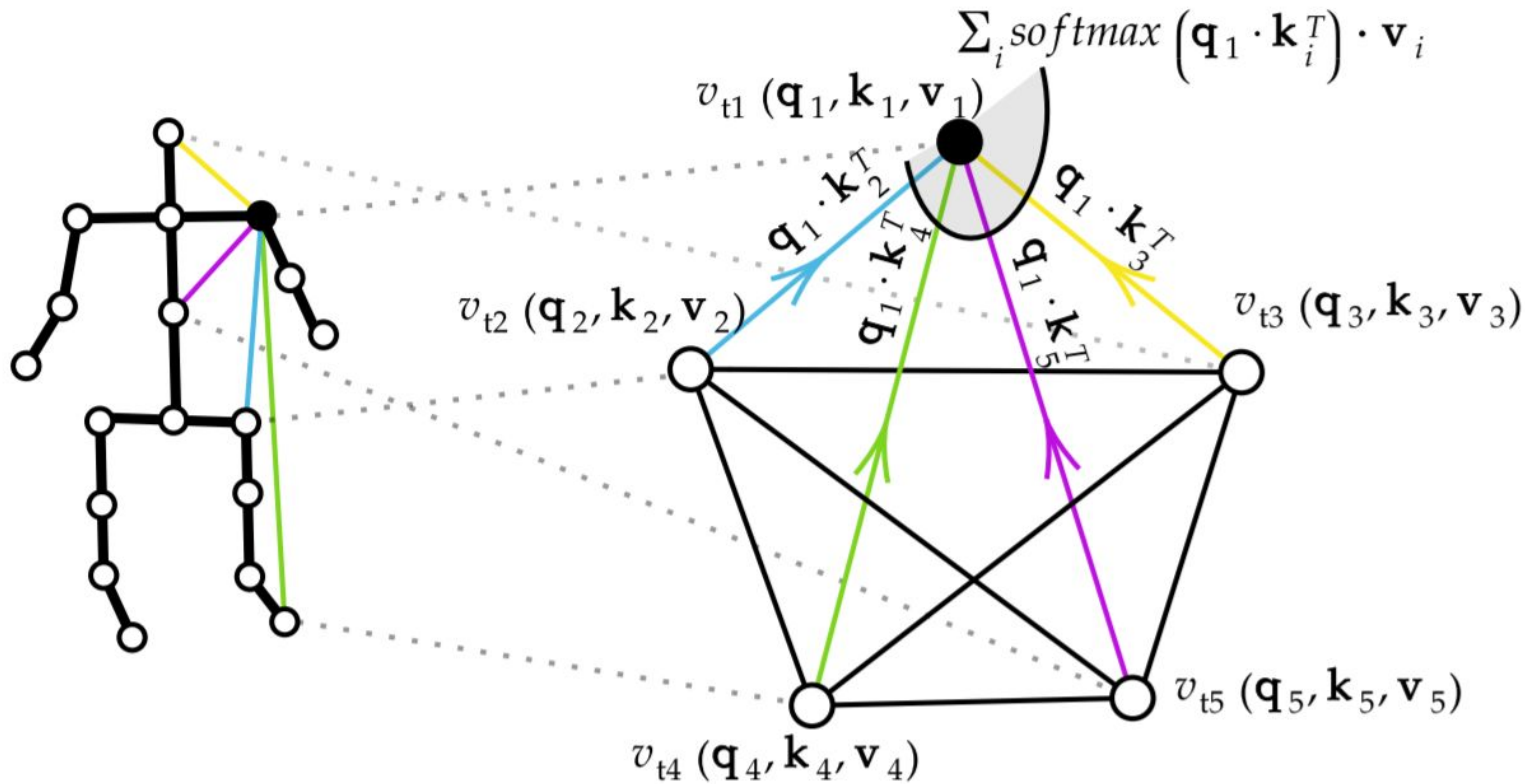
Spatial Transformer (S-TR)

- Spatial Self-Attention (SSA):
  - Applies dot product self-attention within each frame (skeleton)
  - Extracts low-level features capturing relations between body parts
  - Applies multi-headed self attention on features obtained using GCNs

- Spatial Transformer (S-TR) Stream:
  - Temporal Convolutional Network (TCN) applies 2D convolutions with kernel K_t on temporal dimension to obtain final spatial features
  - $\mathrm{S\text{-}TR}(x) = \mathrm{Conv}_{2D(1 \times Kt)}(\mathrm{SSA}(x))$
  - Stacked together to obtain richer features

- Hyperparameters:
  - **Architecture**: 3 x 64 channels + 3 x 128 channels + 3 x 256 channels
  - **Embedding dimension** (key, query, value): 0.25 x C_{out} at each layer
  - **Attention heads**: 8

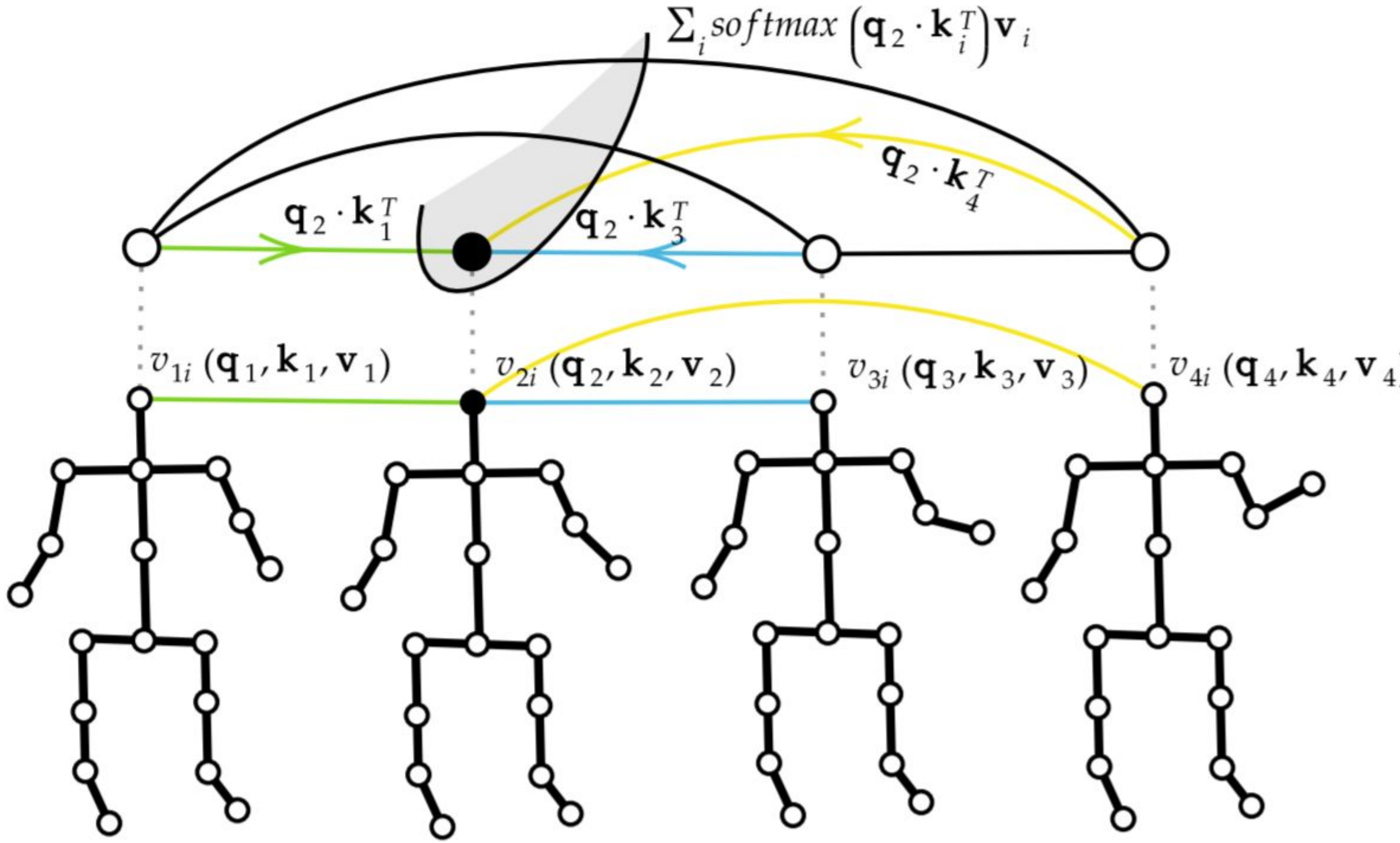
Spatial Self-Attention (SSA) Module



Temporal Transformer (T-TR)

- Temporal Self-Attention (TSA):
  - Applies self-attention across frames (in time)
  - Extracts inter-frame relations between the same nodes across time
  - Applies multi-headed self attention on features obtained using GCNs
- Temporal Transformer (T-TR) Stream:
  - Symmetrical to that of S-TR except V (spatial dim) replaced with T (time)
  - Incorporates MSAF generated contextual embeddings using linear layer
  - `T-TR(x) = TSA(GCN(x), MSAF(x))`
  - Stacked together to obtain richer features
- Hyperparameters:
  - Same as those of S-TR

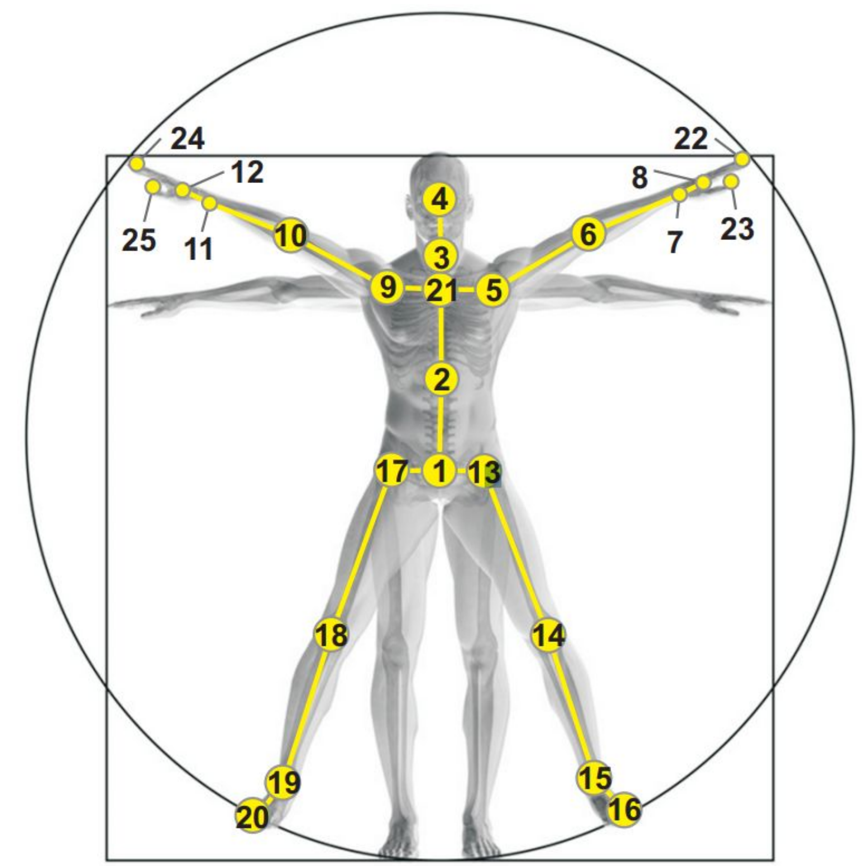
Temporal Self-Attention (TSA) Module



Dataset

- NTU RGB+D 60 dataset [7]:
  - Data collected using a Microsoft Kinect V2
  - Classification among 60 different action classes
  - Largest in-house captured benchmark for 3D human action recognition

- Contains:
  - RGB sequences
  - Depth sequences
  - Infrared sequences
  - Skeleton sequences
    - 25 joints with 3D pose features



- Two benchmarks:
  - Cross-Subject (X-Sub): Split across subjects performing same task
  - Cross-View (X-View): Split across views performing the same task

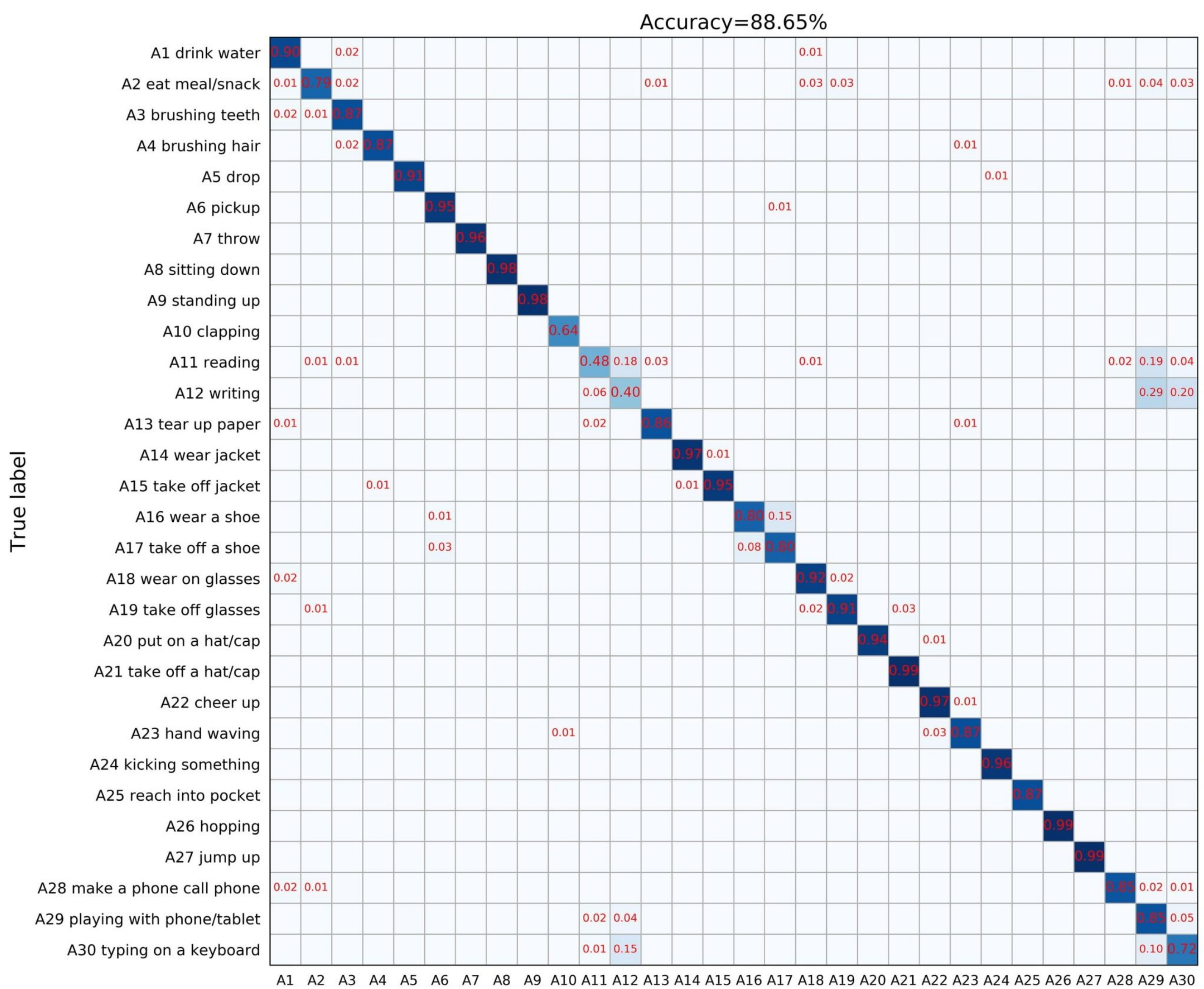
Experimental Setup

- Training ST-CTR:
  - **Framework**: PyTorch
  - **Batch size**: 32
  - **Epochs**: 120
  - **Optimizer**: Stochastic Gradient Descent (SGD)
  - **Initial learning rate**: 0.1
  - **Decay factor**: 0.1 at epochs 60 and 90
  - **Loss**: Cross-entropy

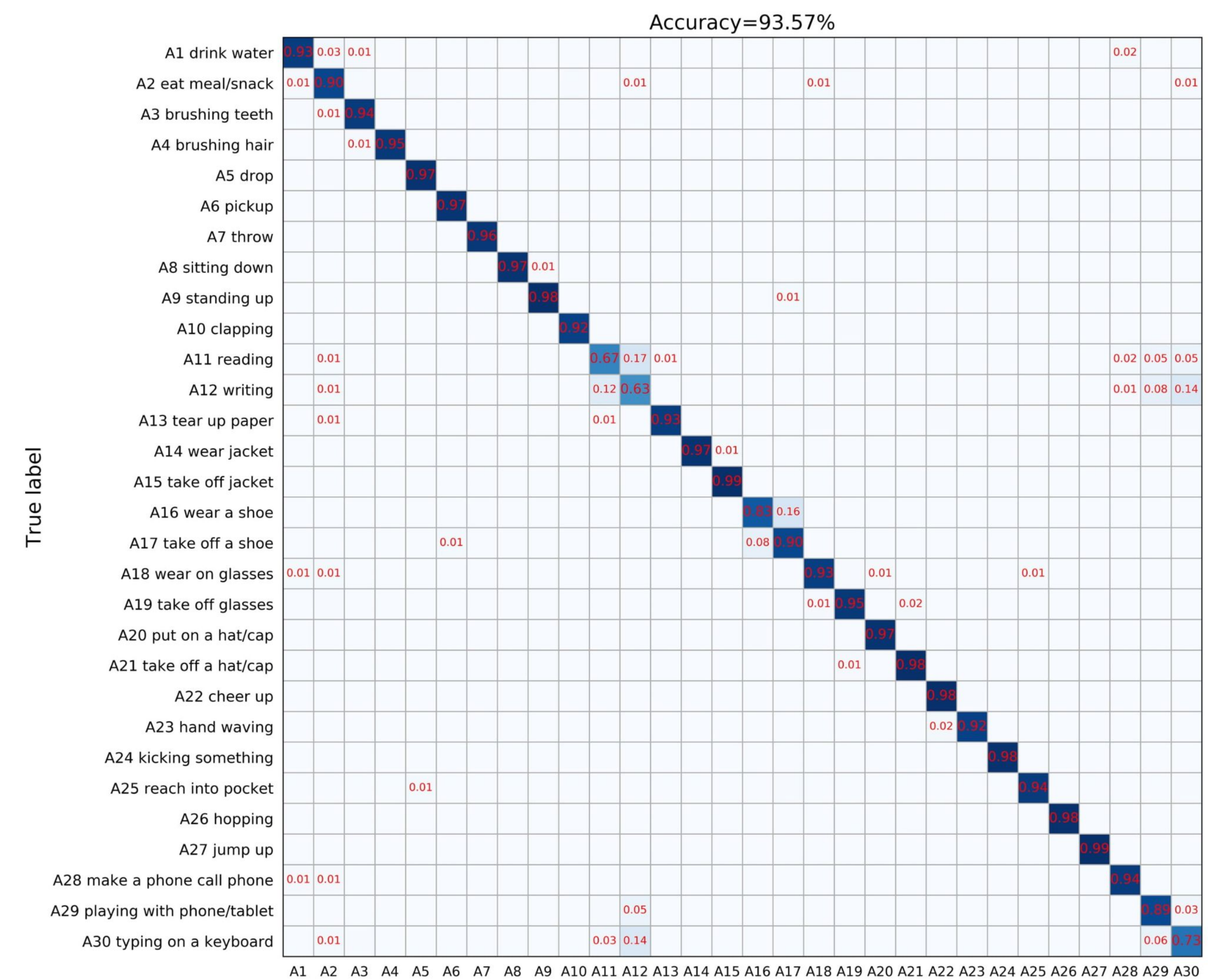
- Regularization:
  - DropAttention for transformers
  - BatchNorm on input joint and video data
  - Global average pooling before softmax layer

Qualitative Results

X-Sub benchmark:



X-View benchmark:



Quantitative Results

Method	X-Sub	X-View
ST-GCN	77.5%	83.3%
PeGCN	85.6%	93.4%
RA-GCN	87.3%	93.5%
PGCN-TCA	88.0%	93.5%
Sem-GCN	86.2%	92.4%
Mix Dimension	87.2%	93.4%
PA-ResGCN-B19	88.5%	93.5%
Dynamic GCN	87.3%	88.6%
ST-TR	85.9%	91.1%
ST-CTR (ours)	**88.7%**	**93.6%**

Table 1. NTU RGB+D 60 test set top-1 classification accuracies

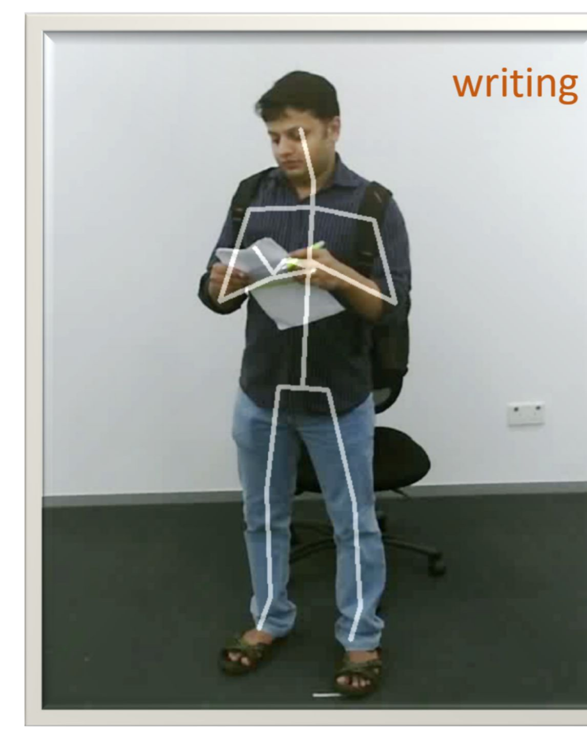
Ablation Study

Components in the pipeline	X-Sub	X-View
S-TR	78.6%	80.7%
T-TR	78.4%	80.5%
MSAF + T-TR	82.1%	85.8%
S-TR + T-TR	85.9%	91.1%
MSAF + S-TR + T-TR (ST-CTR)	**88.7%**	**93.6%**

Table 2. Ablation study of the ST-CTR pipeline

Global Contextual Features

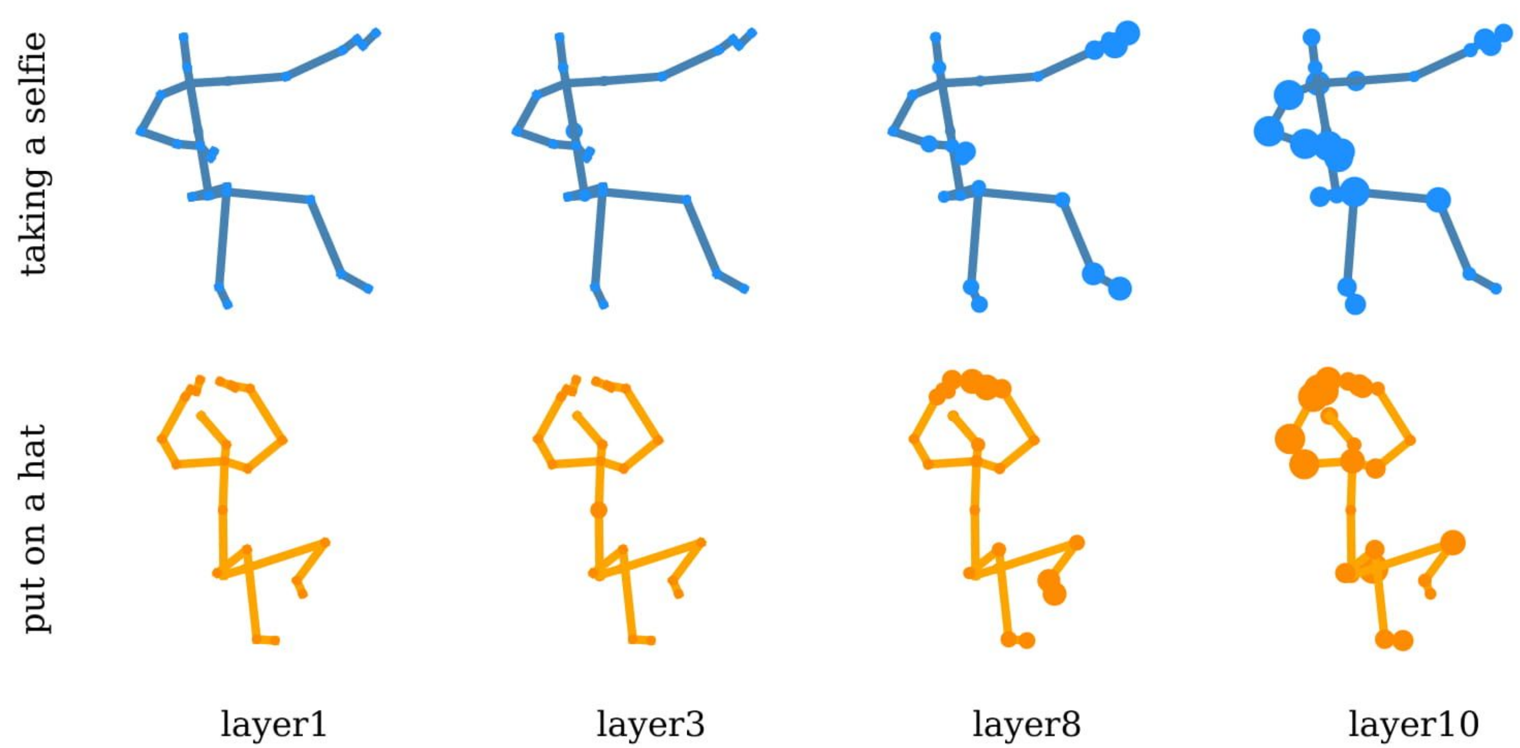
- Sources of errors in other methods:
  - Generalizing to different subjects and views
    - Different height, angle, etc.
    - Need to embed variations in setup as well
  - Actions that are very similar (“reading” vs “writing”)
    - Hard to tell based on skeletons alone
    - Need to embed nuanced interactions with pen and paper into model



- MSAF:
  - Generates feature embeddings using both skeletons (generalize variations) and RGB (embed interactions) frames
  - Allows ST-CTR to outperform other models by incorporating global contextual feature vectors when making decisions

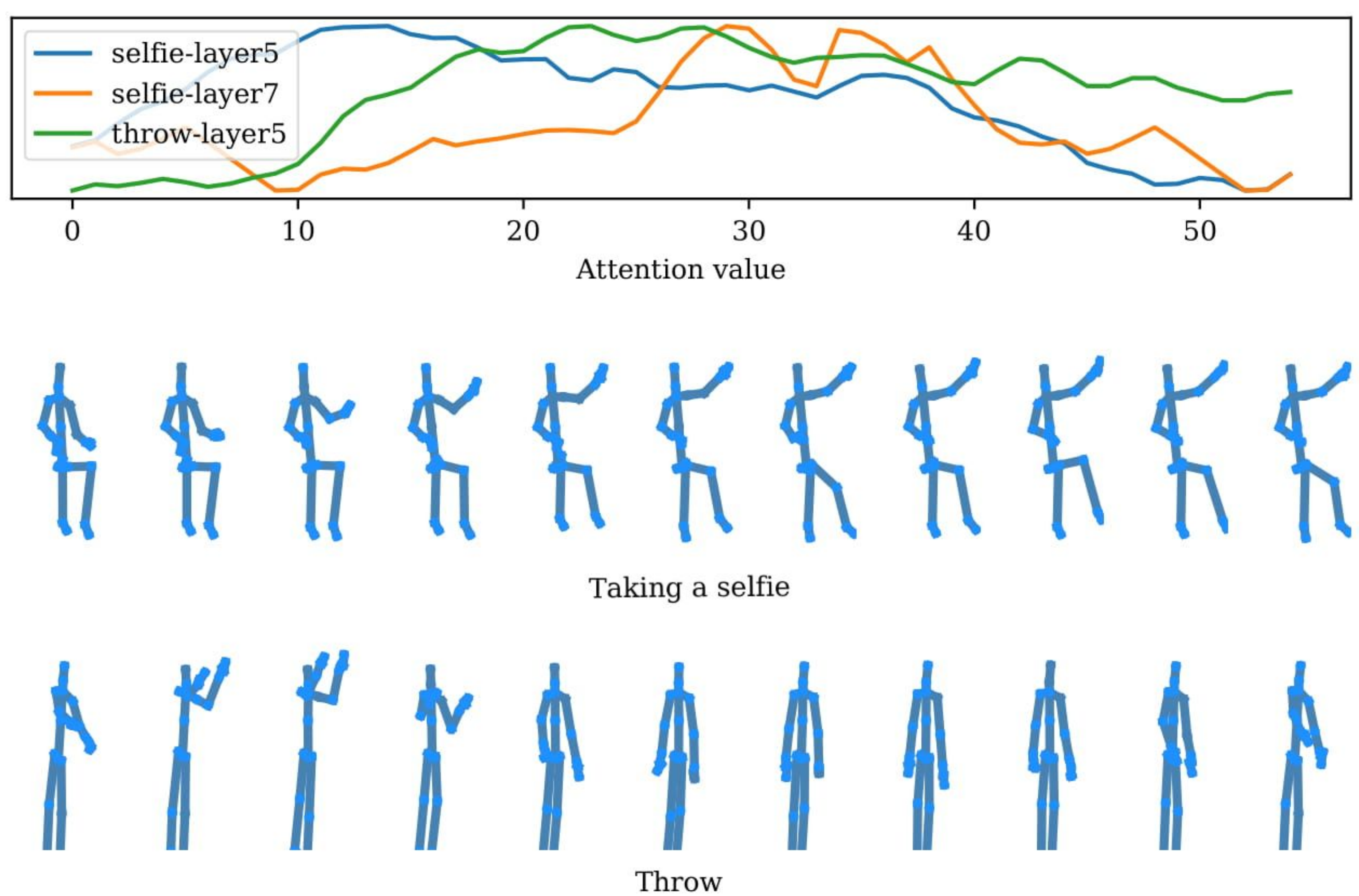
Spatial Self-Attention

- SSA Module:
  - Performs Spatial Self-Attention in S-TR stream
  - Need to focus on joints that are crucial to classifying action
- Spatial attention maps:
  - Node sizes represent importance
  - Less apparent in lower layers since receptive fields are smaller



Temporal Self-Attention

- TSA Module:
  - Performs Temporal Self-Attention in T-TR stream
  - Need to focus on frames across time that are crucial to classifying action
- Temporal attention weights:
  - Frames that convey most information about action have higher weights



Stanford University

Takeaways

- ST-CTR addresses:
  - **Graph Learning Model (GCN)**:
    - Better representation of data to generate richer features
  - **RNN-based Model (Transformer)**:
    - Variable size inputs
    - Can predict actions by generating skeletons for future frames
  - **Contextual Embedding Model (MSAF)**:
    - Variations in setup
    - Actions with subtle differences

Limitations

- Multi-instance action recognition:
  - ST-CTR cannot deal with multiple subjects in the same frame
- Multi-label action recognition:
  - ST-CTR cannot deal with same person performing different actions at the same time
- Video data:
  - ST-CTR uses ground-truth skeletons and cannot perform action recognition on raw RGB video data alone
- Slow inference:
  - ST-CTR needs to compute contextual embeddings from high-resolution images

Future Work

- Areas for future improvement:
    - Better Pose Estimation models to obtain less noisy skeleton data
    - Faster contextual embedding models for fast inference
    - Better regularization methods on graphs (DropEdge, DropGraph, etc.)
    - Action Prediction using generative graph models to generate skeletons from transformer embeddings for future frames

References

[1] Jun Liu, Gang Wang, Ling-Yu Duan, Kamila Abdiyeva, and Alex C Kot. Skeleton-based human action recognition with global context-aware attention lstm networks. IEEE Transactions on Image Processing, 27(4):1586–1599, 2017.

[2] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In Proceedings of the AAAI conference on artificial intelligence, volume 32, 2018.

[3] Chiara Plizzari, Marco Cannici, and Matteo Matteucci. Skeleton-based action recognition via spatial and temporal transformer networks. Computer Vision and Image Understanding, page 103219, 2021.

[4] Lang Su, Chuqing Hu, Guofa Li, and Dongpu Cao. Msaf: Multimodal split attention fusion. arXiv preprint arXiv:2012.07175, 2020.

[5] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 6299–6308, 2017.

[6] Chao Li, Qiaoyong Zhong, Di Xie, and Shiliang Pu. Co-occurrence feature learning from skeleton data for action recognition and detection with hierarchical aggregation. arXiv preprint arXiv:1804.06055, 2018.

[7] Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. Ntu rgb+ d: A large scale dataset for 3d human activity analysis. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1010–1019, 2016.