Robust Human Action Modelling

Research Milestone Report: Oral Presentation

Lei Wang

Australian National University
Data61/CSIRO

February 2, 2023

Australian
National
University

Table of Contents[1]

[1]The presented works were published in TIP'19 (IF:11.041), ICCV'19(A*), TPAMI'20 (IF:24.314), ACMMM'21 (A*), ECCV'22 (A*, oral), ACCV'22 (B, oral, **Best Student Paper Award**). I am grateful to my supervisor **Dr. Piotr Koniusz** (Data61/CSIRO&ANU).

Introduction: Action Recognition & A Comparative Review

Introduction: Action Recognition & A Comparative Review

- **Action Recognition**: recognize/identify actions in video
- Motivations: many useful **applications**
- Challenging **problems**: viewpoints, partial occlusion & self-occlusion, *etc.*
- **Datasets** & **protocols**:

Table 1: Some benchmarks for 3D action recognition.

Datasets	Year	Classes	Subjects	#Views	#videos	Sensor	Modalities	#joints
UWA3D Activity	2014	30	10	1	701	Kinect v1	RGB + Depth + 3DJoints	15
UWA3D Multiview Activity II	2015	30	9	4	1,070	Kinect v1	RGB + Depth + 3DJoints	15
Charades	2016	157	-	-	66,500	-	RGB	-
NTU RGB+D	2016	60	40	80	56,880	Kinect v2	RGB + Depth + IR + 3DJoints	25
NTU RGB+D 120	2019	120	106	155	114,480	Kinect v2	RGB + Depth + IR + 3DJoints	25
Kinetics-skeleton	2019	400	-	-	260,232	-	2DJoints	18
Kinetics-700	2020	700	-	-	647,907	-	RGB	-

  - cross-subject/single-view & cross-view action recognition
  - zero-, one- & few-shot action recognition
- **Techniques**:
  - RGB videos (*e.g.*, IDT, Two-stream network, C3D, TSN, I3D, *etc.*)
  - Depth videos (*e.g.*, HON4D, HOPC, *etc.*)
  - Skeleton sequences (*e.g.*, ST-GCN *etc.*)
- A comparative **review**[2] of recent action recognition algorithms

[2] Wang, L., Huynh, D. Q., & Koniusz, P. (2020). **A comparative review of recent kinect-based action recognition algorithms**. *IEEE TIP*, 29, 15-28.

Video-based Action Recognition

Motivation, key ideas & pipeline

Motivation:

- Transition: handcrafted feature → CNN models
- Handcrafted features:
  - capture **domain specific information**
  - encoded with Bag-of-Words (BoW) / Fisher Vectors (FV)
  - **fused with CNNs** for better performance but **costly**

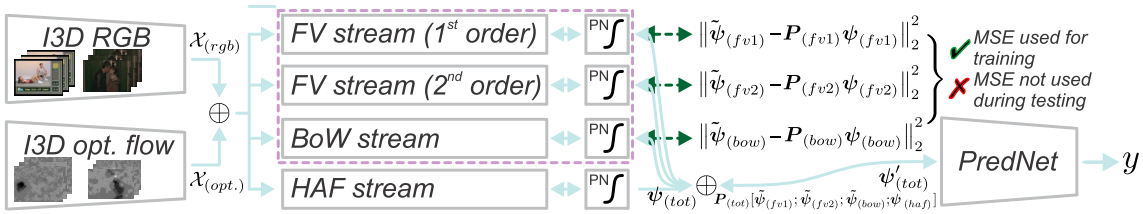


Figure 1: The overview of our pipeline.

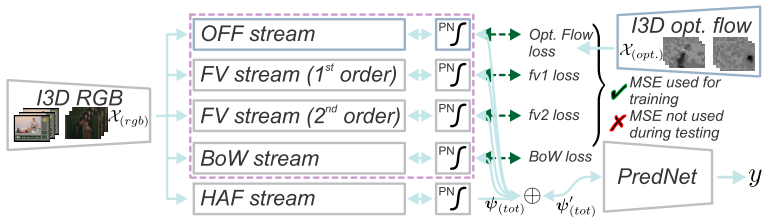


Figure 2: Hallucinating OFF.

Our model:

- learn to 'translate' the CNN output (*e.g.*, I3D) to IDT-based BoW/FV
- even 'translate' the CNN output to I3D Optical Flow Features (OFF)
- Test stage:
  - BoW, FV & OFF streams hallucinate global descriptors
    - remove the need of actually computing IDT/OFF
    - simplify the action recognition pipeline

Results and Discussions

	sp1	*sp2*	*sp3*	*sp4*	*sp5*	*sp6*	*sp7*	mAP
HAF+BoW halluc.	73.9	71.6	76.2	70.7	76.3	71.9	63.4	71.9%
HAF+BoW halluc.+SK/PN	73.9	75.8	72.2	73.9	77.0	73.6	68.8	**73.6%**
HAF* only	74.6	73.2	77.0	75.1	76.1	75.6	71.9	74.8%
HAF*+BoW halluc.	78.8	75.0	84.1	76.0	77.0	78.3	75.2	**77.8%**
HAF*+BoW hal.+MSK/PN	80.1	79.2	84.8	83.9	80.9	78.5	75.5	**80.4%**
HAF•+BoW hal.+MSK/PN	80.8	80.9	85.0	83.9	82.0	79.8	79.6	**81.7%**
ditto+OFF hal.	81.2	81.2	84.9	83.4	84.2	78.9	79.1	**81.8%**
I3D+BoW MTL•	79.1	78.1	83.6	78.7	79.1	78.6	76.5	79.1%
KRP-FS 70.0%	KRP-FS+IDT 76.1%		GRP 68.4%		GRP+IDT 75.5%			

Table 2: Evaluations of (*top*) our methods and (*bottom*) comparisons to the state of the art on MPII.

Why our pipeline works well?

- enforce a CNN to learn IDT / IDT is unlikely to be captured by CNNs
- co-regularize I3D resembles domain adaptation
- multi-task learning (MTL) boosts generalization & prevents overfitting (task specific losses)
- self-supervision

With state-of-the-art results, we hope our method (DEEP-HAL[3]) will **spark a renewed interest** in IDT-like descriptors.

[3]Wang, L., Koniusz, P., & Huynh, D. Q. (2019). **Hallucinating idt descriptors and i3d optical flow features for action recognition with cnns**. In *ICCV* (pp. 8698-8708).

Motivation

- build on a concept of **self-supervision**
- learn to predict both action concepts and auxiliary descriptors
- motivate the use of **higher-order statistics**
  - capture 4 **statistical moments**: mean, covariance, coskewness & cokurtotsis
  - describe the covariance matrix by its leading n' eigenvectors (**subspace**)

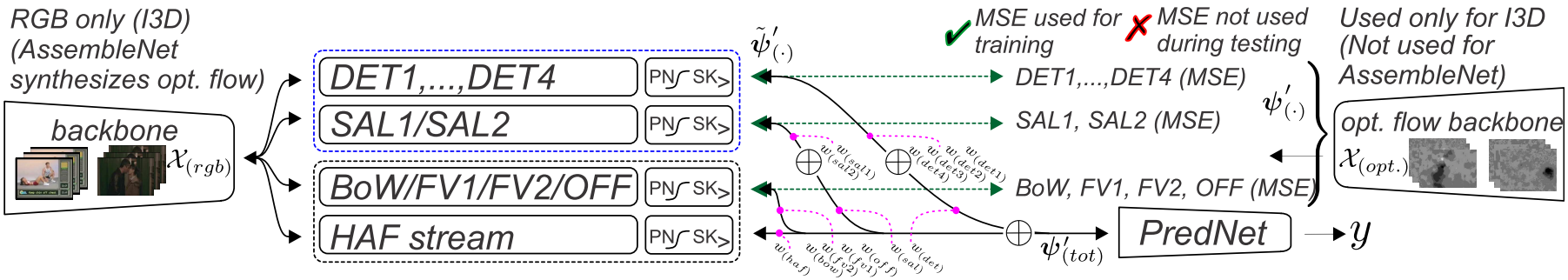
Figure 3: We use detectors & saliency in hallucination descriptors. 3a:bounding boxes from Inception V2, Inception ResNet V2, *ResNet101* & NASNet. 3b: MNL saliency detector[4] focuses on spatial regions (region-wise saliency). 3c: ACLNet saliency detector[5] discovers motion regions (temporal saliency).

[4]Zhang, J., Zhang, T., Dai, Y., Harandi, M., & Hartley, R. (2018). **Deep unsupervised saliency detection: A multiple noisy labeling perspective**. In *CVPR* (pp. 9029-9038).

[5]W. Wang, J. Shen, J. Xie, M. -M. Cheng, H. Ling and A. Borji (2021). **Revisiting Video Saliency Prediction in the Deep Learning Era**. In *IEEE TPAMI* (pp. 220-237).

ODF, SDF & the pipeline

- Object Detection Features (ODF)
  - one-hot detection, ImageNet scores, embeded confidence scores of detection, embedded bounding box coordinates & embedded normalized frame index
  - stack per bounding box per frame features into a matrix
  - extract the mean, leading eigenvectors of covariance, skewness & kurtosis
- Saliency Detection Features (SDF)
  - kernelized descriptor on spatio-angular gradient distribution of saliency maps & intensity patterns
- An ODF per detector & an SDF per saliency detector



- The OFF stream is supervised by $\mathcal{X}_{(opt.)}$.
- *DET1*, ...,*DET4* & *SAL1*/*SAL2* corresponding to ODF & SDF (dashed blue).

Results and Discussions

HAF/BoW/FV hal.	*DEEP-HAL+ W+G+ODF (SK512)*	*DEEP-HAL+ W+G+SDF (SK512)*
43.1	47.22	45.30

DEEP-HAL+W+G+ ODF+SDF (SK512)	*DEEP-HAL+W+G+ ODF+SDF (SK1024)*	*DEEP-HAL+W+G+ ODF+SDF (exact)*
49.06	**50.14**	**50.16**

Table 3: Evaluations on Charades (I3D backbone).

AssembleNet++ 50 (Kinetics-400 pre-training)			
baseline	*ODF+SDF (SK512)*	*ODF+SDF (SK1024)*	*ODF+SDF (exact)*
53.8	55.81	**56.94**	**57.30**

AssembleNet++ 50 (without pre-training)			
baseline	*ODF+SDF (SK512)*	*ODF+SDF (SK1024)*	*ODF+SDF (exact)*
56.7	60.71	**61.98**	**62.29**

Table 4: Eval. on Charades (AssembleNet++).

For more details, please refer to our paper[6].

Discussions:

- a large margin of performance gain
  - detection/saliency features boosts results by $\sim\mathbf{6}\%$
  - ODF and SDF are highly complementary
- a simple approach
  - lightweight by comparison
  - save computational time
  - 'orthogonal' to backbones

[6] Wang, L., & Koniusz, P. (2021). **Self-supervising action recognition by statistical moment and subspace descriptors**. In *ACMMM* (pp. 4324-4333).

Skeleton-based Action Recognition

Overview

We are interested in matching pairs of temporal sequences (or time series) for few-shot learning, time series completion and classification.

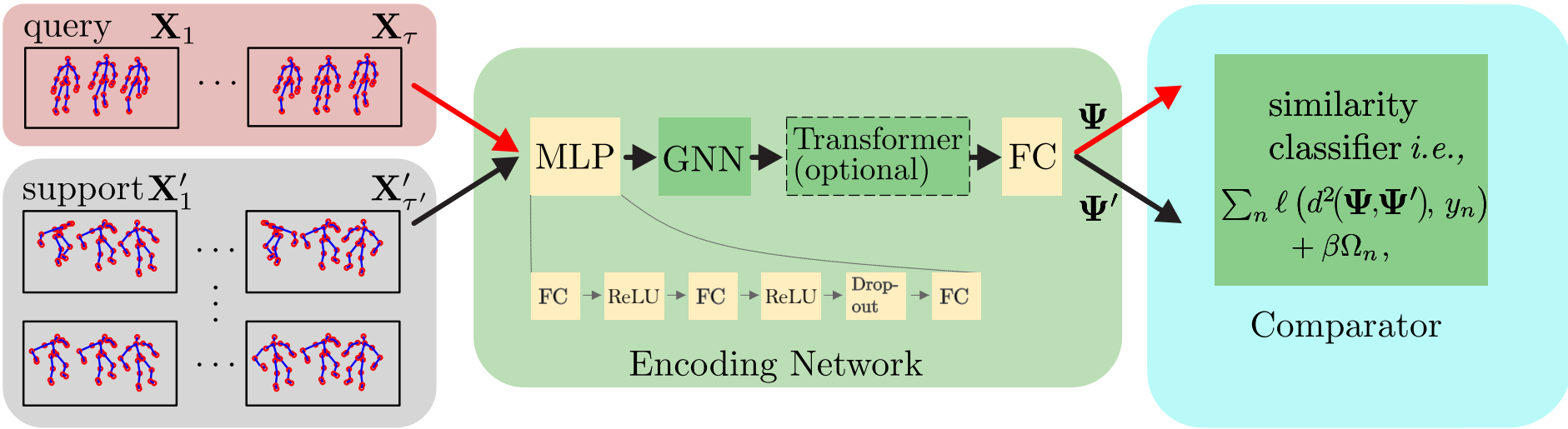


Figure 4: Example few-shot action recognition pipeline.

- We train the Encoding Network.
- The comparator learns the notion of similarity between query-support pairs.
- At the test time, given a set of support sequences with labels, we can decide which one matches the query.
- The empirical loss $\ell(\cdot)$ is encouraged to reach 0 if query-support pair has the same class labels. For pairs with non-matching labels, $\ell(\cdot)$ is encouraged to be large.

Overview

Formally, similarity learning minimizes the empirical loss $\ell(\cdot)$ and some regularization term $\Omega(\cdot,\cdot)$ expressing our belief about the model:

$$\sum_n \ell(d^2(\mathbf{\Psi}_n, \mathbf{\Psi}'_n), y_n) + \beta\Omega_n(\mathbf{\Psi}_n, \mathbf{\Psi}'_n).$$

Query: $\mathbf{\Psi} \equiv [\boldsymbol{\psi}_1, ..., \boldsymbol{\psi}_\tau]$ with τ temporal frames (or blocks).
Support: $\mathbf{\Psi}' \equiv [\boldsymbol{\psi}'_1, ..., \boldsymbol{\psi}'_{\tau'}]$ with τ' temporal frames (or blocks).

However, distance $d(\cdot,\cdot)$ is suboptimal for matching temporal sequences:

- Temporal location and speed of actions vary.
- Temporal patterns within the same class have high intra-class variance: no two sequences are identical.
- Same actors never perform the same action exactly the same way.
- So-called (Soft-)Dynamic Time Warping (DTW) overcomes the above issues[7]. We build on it.

[7] Cuturi, M., & Blondel, M. (2017, July). **Soft-dtw: a differentiable loss function for time-series**. In *ICML* (pp. 894-903). PMLR.

Motivation

Compare the Euclidean distance *vs.* (Soft-)Dynamic Time Warping (DTW):

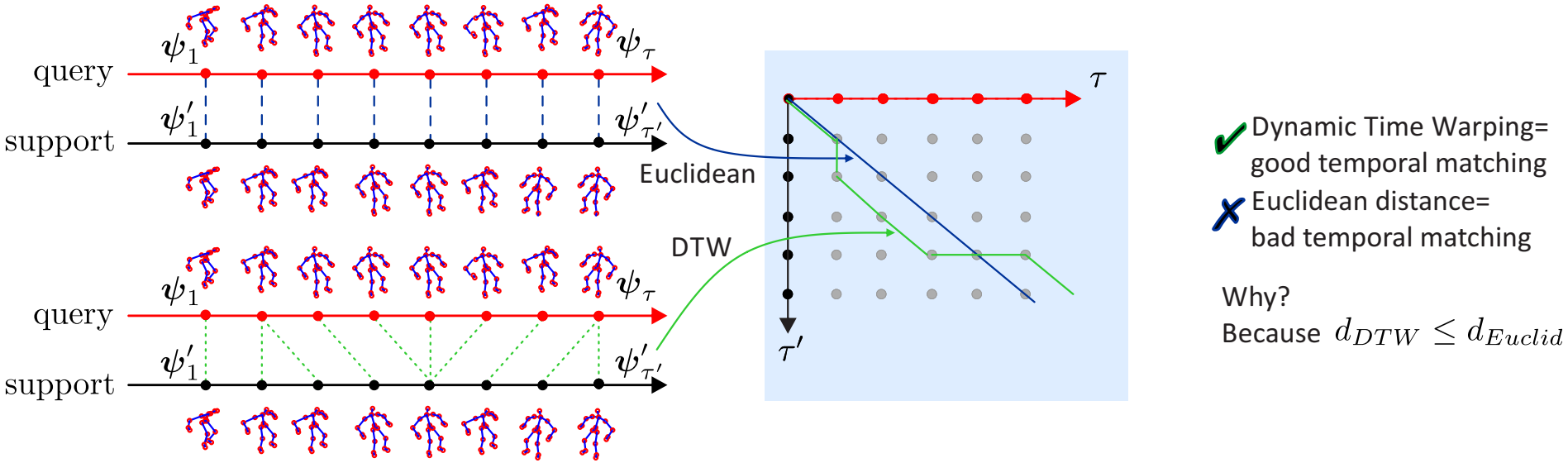


Figure 5: Euclidean dist. (top) *vs.* DTW (bottom). Corresponding matching paths (right).

- The Euclidean distance naively compares features of corresponding frames of two sequences Ψ and Ψ'. See support-query matching of frames (top plot).
- The (Soft-)Dynamic Time Warping (bottom) is able to match better human poses taking into account temporal variations.
- DTW performs that 'better' matching (see the green matching path on the right) by factoring out temporal variations. The black path is suboptimal.

Motivation

However, sequences Ψ and Ψ' suffer from **the observation noise**.
Compare uncertainty-DTW *vs.* soft-DTW under the noise (indicated in gray):

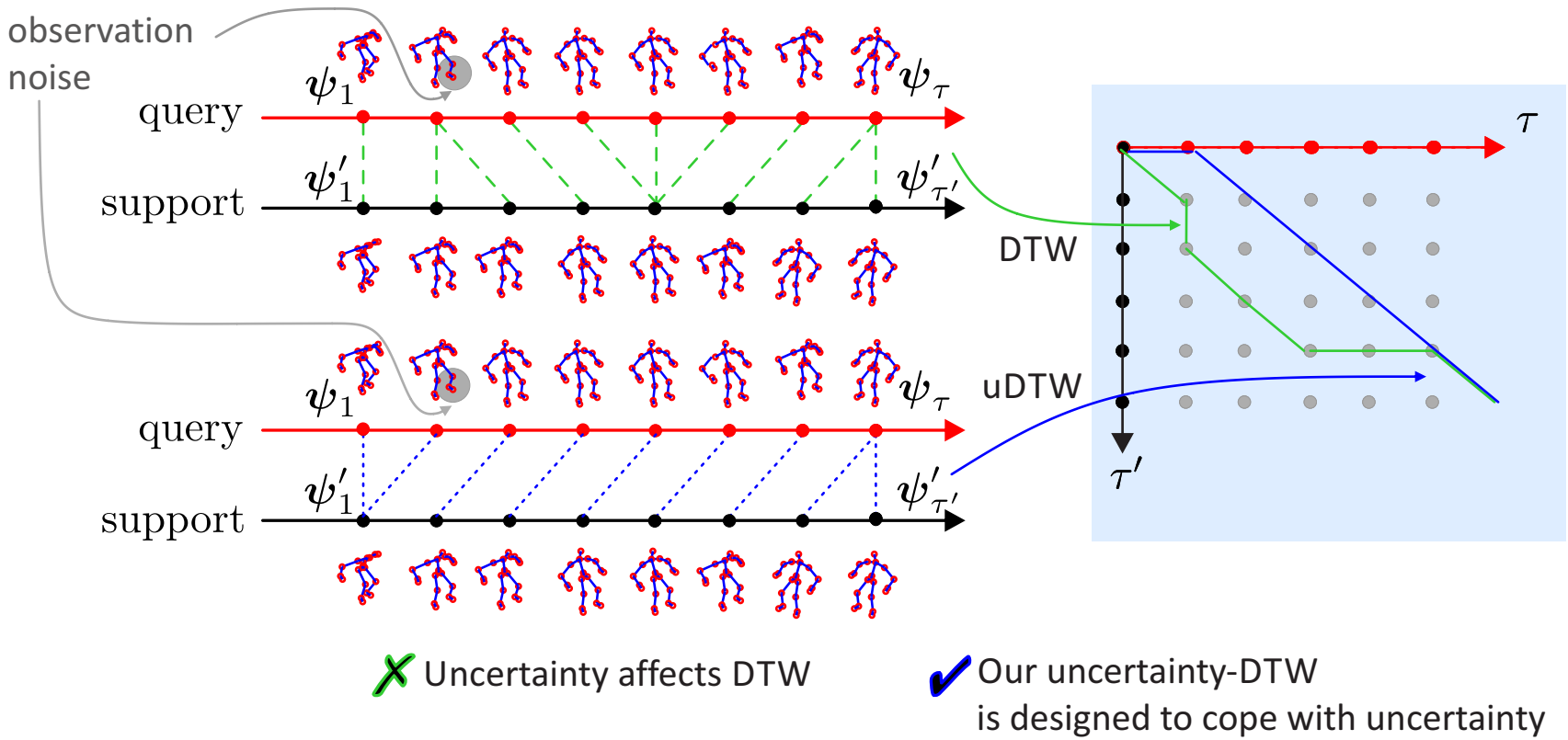


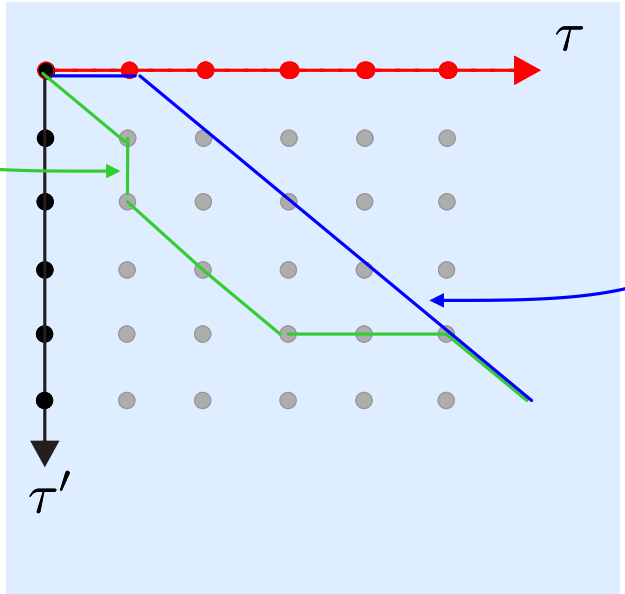
Figure 6: Soft-DTW. (top) *vs.* uncertainty-DTW (bottom).

- Blue path (right) takes uncertainty into account; green path does not.
- Thus, the blue path provides more robust distance for similarity learning.

Approach

$\sum_t d^2\left(\boldsymbol{\psi}_{n(t)}, \boldsymbol{\psi}'_{m(t)}\right)$

DTW
(distance accumulated along the path)



$\sum_t \frac{1}{2\sigma^2_{n(t),m(t)}} d^2\left(\boldsymbol{\psi}_{n(t)}, \boldsymbol{\psi}'_{m(t)}\right)$

$\Omega = \sum_t \log \sigma_{n(t),m(t)}$

uDTW
(uncertainty-weighted distance accumulated along the path)

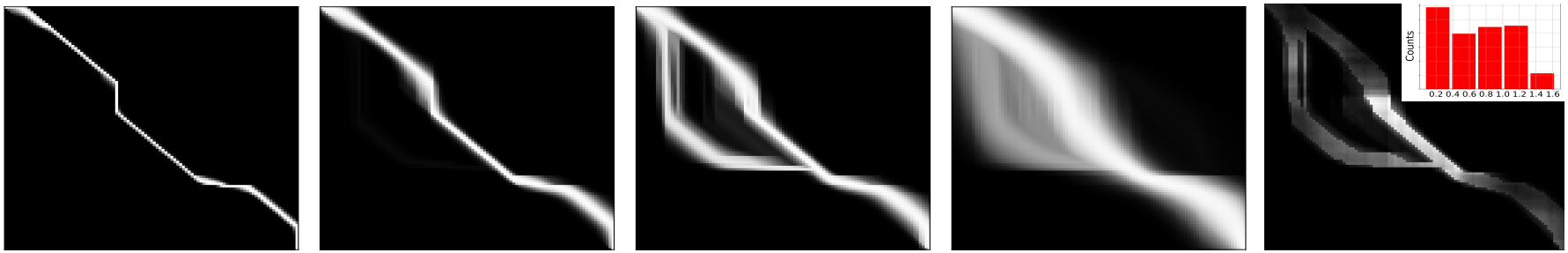
$\Omega =$ uncertainty penalty
(regularization accumulated along the path)

Figure 7: Soft-DTW *vs.* uncertainty-DTW.

- Uncertainty-DTW models the uncertainty for each frame (or temporal block).
- Each path is a solution to the Maximum Likelihood Estimation: each node on the path is described by the Gaussian with variance.
- MLE 'explains' the distances on the path by the modelled distribution.
- Log-likelihood results in d_{uDTW} (see derivations in the paper).
- Additionally, Ω is penalty for selecting (trivially) large uncertainty.

Approach

Our **uncertainty-DTW** can capture 'alternative' paths:



(a) sDTW$_{\gamma=0.01}$ (b) sDTW$_{\gamma=0.1}$ (c) uDTW$_{\gamma=0.01}$ (d) uDTW$_{\gamma=0.1}$ (e) uncertainty

Figure 8: With higher γ controlling softness, in (b) & (d) more paths become 'active'. In (c) & (d), uDTW has two possible routes due to uncertainty modeling.

- Soft-DTW (plots (a) & (b)) produces single paths ('fuzziness is due to soft-maximum operator selecting the best path).
- Uncertainty-DTW (plots (c) & (d)) produces alternative paths merging where the uncertainty $\sigma_{n,m}$ (plot (e)) is large.
- $\sigma_{n,m}$ is obtained from a small MLP called SigmaNet (we have observed it is better to optimize over SigmaNet parameters than directly over $\sigma_{n,m}$.

Pipeline: Supervised Few-shot Action Recognition

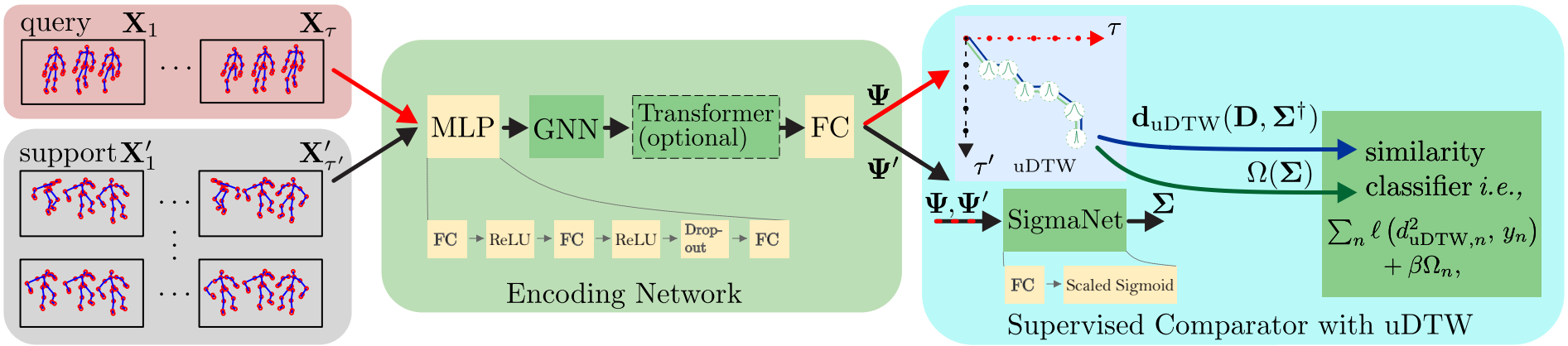


Figure 9: Supervised few-shot action recognition with the uncertainty-DTW (uDTW).

Our model contains:

- Encoding Network (backbone); each sequence is split into temporal blocks.
- Comparator has access to each temporal block features $\boldsymbol{\psi}_1, ..., \boldsymbol{\psi}_\tau$ and $\boldsymbol{\psi}'_1, ..., \boldsymbol{\psi}'_{\tau'}$ of query-support pairs.
- SigmaNet produces the uncertainty variable $\mathbf{\Sigma}$
- The objective function is a trade-off between the empirical loss $\ell(\cdot)$ with uncertainty-DTW and the uncertainty penalty (regularization) $\Omega(\cdot)$.

Pipeline: Unsupervised Few-shot Action Recognition

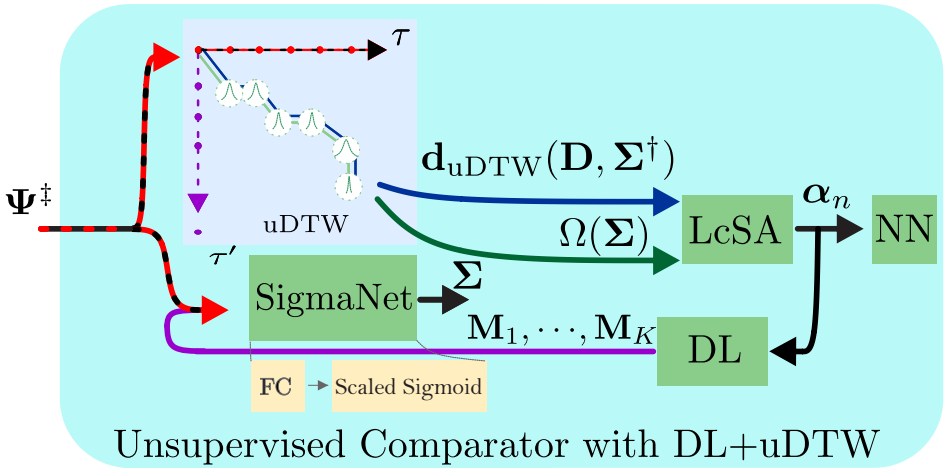


Figure 10: Unsupervised few-shot action recognition with the uncertainty-DTW (uDTW).

- We train Encoding Network (backbone) but in an unsupervised manner.
- Comparator learns a dictionary (DL) which contains 'abstract' dictionary sequences (clusters).
- LcSA is an encoder of sequences into the dictionary space.
- Interaction between LcSA encoder and dictionary can be thought as soft clustering that uses the uncertainty-DTW distance.
- At the test time, the nearest neighbor on encoded sequences is used to match support sequence (known labels) with the query (unknown label).

Pipeline: Forecasting the Evolution of Time Series

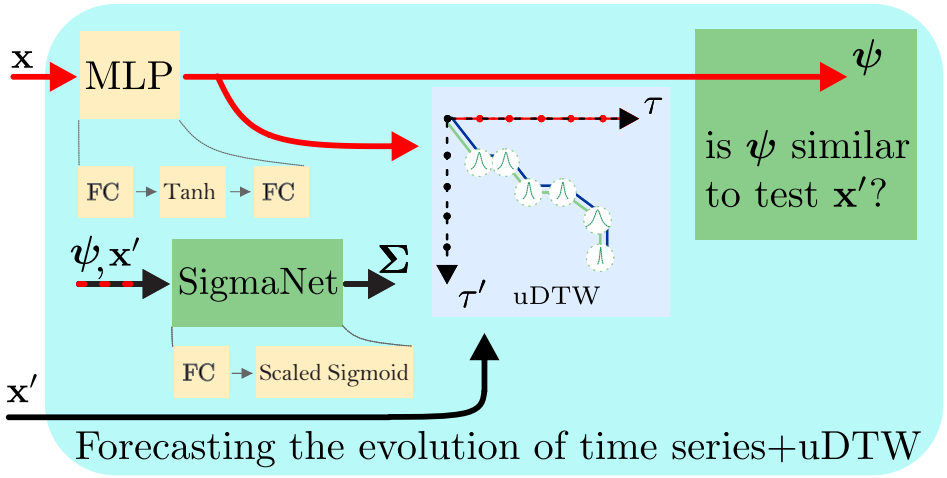
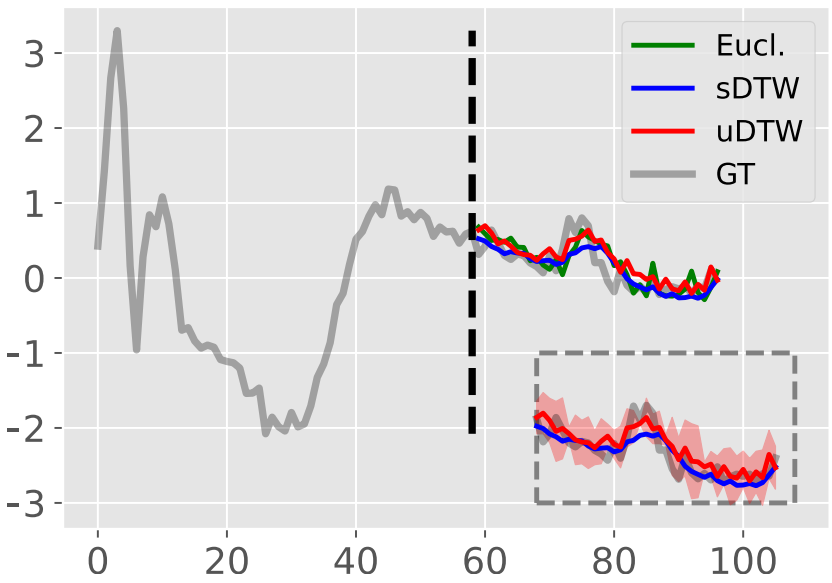


Figure 11: Predicting Evolution of Time Series.

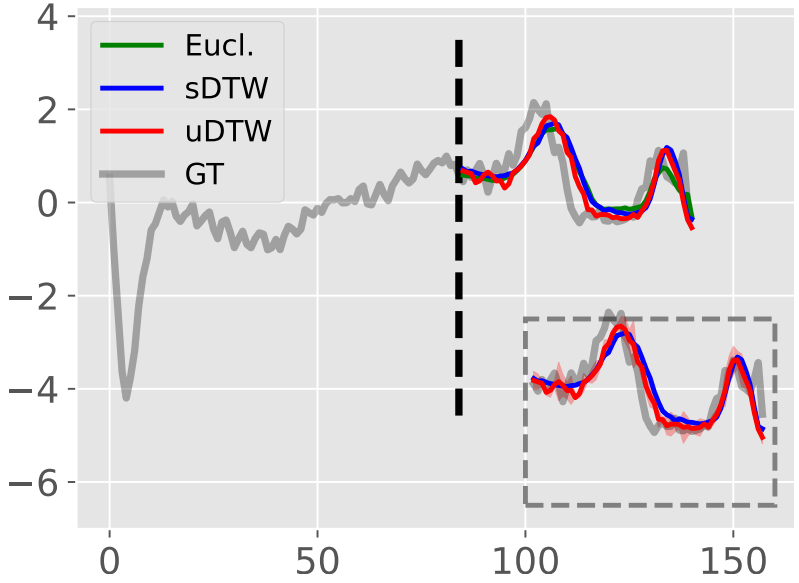
- Variable $\mathbf{x}$ is the first half of time series, and $\mathbf{x}'$ is the second half of time series.
- MLP learns to predict $\mathbf{x}'$ with MLP+uncertainty-DTW from $\mathbf{x}$.

Results: Forecasting the Evolution of Time Series

- Given the first part of a time series, we
  - train 3 multi-layer perception (MLP) to predict the remaining part
  - use the Euclidean, sDTW or uDTW distance per MLP



(a) ECG200



(b) ECG5000

Figure 12: We use ECG200 and ECG5000 in UCR archive, and display the prediction obtained for the given test sample and the ground truth (GT). Oftentimes, we observe that uDTW helps predict the sudden changes well.

Results: Few-shot Action Recognition

For more details, results and discussions, please refer to our paper [8].

Table 5: Evaluations on NTU-60.

#classes	10	20	30	40	50
Supervised					
MatchNets	46.1	48.6	53.3	56.3	58.8
ProtoNet	47.2	51.1	54.3	58.9	63.0
TAP	54.2	57.3	61.7	64.7	68.3
Euclidean	38.5	42.2	45.1	48.3	50.9
sDTW	53.7	56.2	60.0	63.9	67.8
sDTW div.	54.0	57.3	62.1	65.7	69.0
uDTW	56.9	61.2	64.8	68.3	72.4
Unsupervised					
Euclidean	20.9	23.7	26.3	30.0	33.1
sDTW	35.6	45.2	53.3	56.7	61.7
sDTW div.	36.0	46.1	54.0	57.2	62.0
uDTW	37.0	48.3	55.3	58.0	63.3

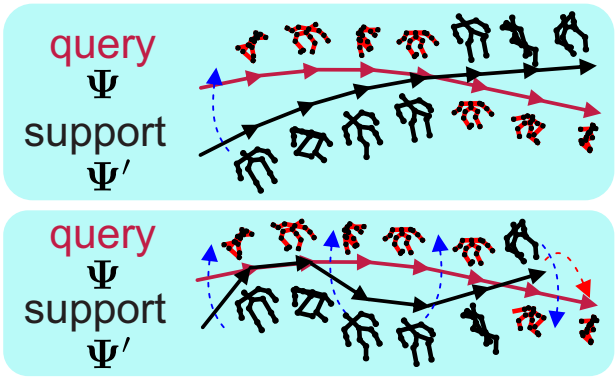
Table 6: Evaluations on NTU-120.

#classes	20	40	60	80	100
Supervised					
MatchNets	20.5	23.4	25.1	28.7	30.0
ProtoNet	21.7	24.0	25.9	29.2	32.1
TAP	31.2	37.7	40.9	44.5	47.3
Euclidean	18.7	21.3	24.9	27.5	30.0
sDTW	30.3	37.2	39.7	44.0	46.8
sDTW div.	30.8	38.1	40.0	44.7	47.3
uDTW	32.2	39.0	41.2	45.3	49.0
Unsupervised					
Euclidean	13.5	16.3	20.0	24.9	26.2
sDTW	20.1	25.3	32.0	36.9	40.9
sDTW div.	20.8	26.0	33.2	37.5	42.3
uDTW	22.7	28.3	35.9	39.4	44.0

sDTW div.: Blondel *et al.*, **Differentiable divergences between time series**. *AISTATS 2021*.
TAP: Bing Su & Ji-Rong Wen, **Temporal Alignment Prediction for Supervised Representation Learning and Few-Shot Sequence Classification**, *ICLR 2022*.

[8] Wang, L., & Koniusz, P. (2022). **Uncertainty-DTW for Time Series and Sequences**. In *ECCV*, oral.

Motivation



Matching query-support features under varying viewpoints of 3D poses:

- (*top*) rotate a support trajectory onto a query trajectory (naive).
- (*bottom*) advanced viewpoint alignment strategy is needed: locally follow complicated non-linear paths but **assume viewpoints change smoothly in time**, *e.g.*, no large abrupt changes along the path.

To **learn similarity/dissimilarity** between pairs of query-support sequences:

- find a smooth joint viewpoint-temporal alignment.
- minimize/maximize d_{JEANIE} for same/different support-query labels.

A viewpoint invariant distance can be defined as:

$$d_{\text{inv}}(\mathbf{\Psi}, \mathbf{\Psi}') = \inf_{\gamma,\gamma' \in T} d\big(\gamma(\mathbf{\Psi}), \gamma'(\mathbf{\Psi}')\big), \tag{1}$$

- T is a set of transformations required to achieve a viewpoint invariance.
- T may include 3D rotations to rotate one trajectory onto the other (or each 3D pose onto the corresponding 3D pose).
- Such global viewpoint alignment of two sequences or local alignment of 3D poses are **suboptimal**. T may realise better transformation strategies...

Thus, we propose a FSAR approach that learns on skeleton-based 3D body joints by **Joint tEmporal and cAmera viewpoiNt alIgnmEnt (JEANIE)**.

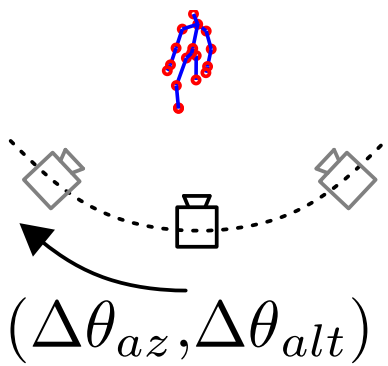
JEANIE

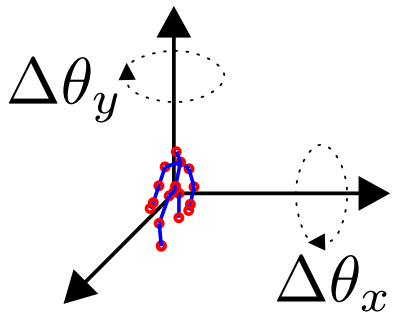
Sequences that are being matched might have been captured under different camera viewpoints or subjects might have followed different trajectories.
Thus, to model 3D pose variations, we:

- exploit the **projective camera geometry**.
- propose **the smooth path** in DTW should **simultaneously perform temporal & viewpoint alignment**

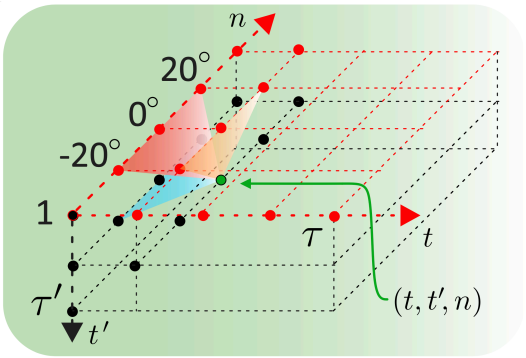
JEANIE has the transportation plan $\mathcal{A}'$where apart of steps $\downarrow$, $\searrow$, $\rightarrow$ for temporal axes (indicated as τ and τ'), JEANIE can also take **additional steps on the viewpoint axis, *e.g.*, step inward, inward-down, *etc.***.
Thus, apart from temporal block counts τ (query) & τ' (support), for query sequences we simulate $K=2\eta_{az}+1$, $K'=2\eta_{alt}+1$ camera viewpoints (or Euler angles). We have:

- possible η_{az} left and η_{az} right steps from the **initial camera azimuth**,
- and η_{alt} up and η_{alt} down steps from the **initial camera altitude**.





JEANIE (cont.)



JEANIE is given as:

$$d_{\text{JEANIE}}(\boldsymbol{\Psi}, \boldsymbol{\Psi}') = \underset{\mathbf{A}\in\mathcal{A}'}{\text{SoftMin}_\gamma} \left\langle \mathbf{A}, \boldsymbol{\mathcal{D}}(\boldsymbol{\Psi}, \boldsymbol{\Psi}') \right\rangle, \qquad (2)$$

where $\boldsymbol{\mathcal{D}} \in \mathbb{R}_+^{K\times K'\times\tau\times\tau'} \equiv \left[d_{\text{base}}(\boldsymbol{\psi}_{m,k,k'}, \boldsymbol{\psi}'_n)\right]_{\substack{(m,n)\in\mathcal{I}_\tau\times\mathcal{I}_{\tau'} \\ (k,k')\in\mathcal{I}_K\times\mathcal{I}_{K'}}}$.

Algorithm 1 Joint tEmporal and cAmera viewpoiNt alIgnmEnt (JEANIE).

Input (forward pass): $\boldsymbol{\Psi}, \boldsymbol{\Psi}'$, $\gamma > 0$, $d_{\text{base}}(\cdot,\cdot)$, ι-max shift.

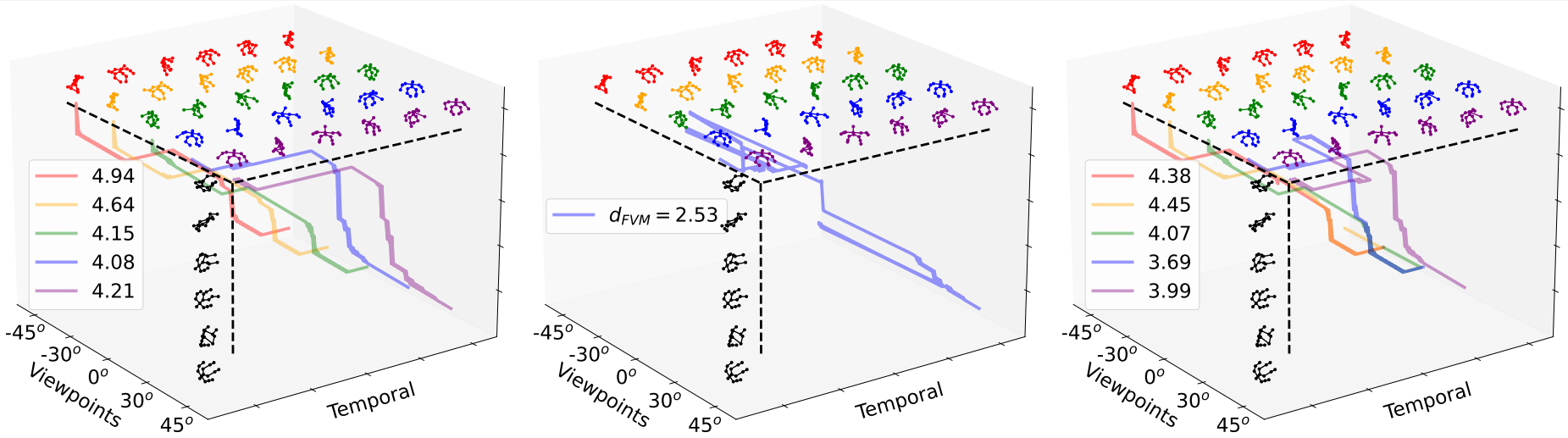
$r_{:,:,:} = \infty$, $r_{n,1,1} = d_{\text{base}}(\boldsymbol{\psi}_{n,1}, \boldsymbol{\psi}'_1)$, $\forall n \in \{-\eta, ..., \eta\}$
$\Pi \equiv \{-\iota, ..., 0, ..., \iota\} \times \{(0,1), (1,0), (1,1)\}$
for $t \in \mathcal{I}_\tau$:
  for $t' \in \mathcal{I}_{\tau'}$:
    if $t \neq 1$ or $t' \neq 1$:
      for $n \in \{-\eta, ..., \eta\}$:
        $r_{n,t,t'} = d_{\text{base}}(\boldsymbol{\psi}_{n,t}, \boldsymbol{\psi}'_{t'}) + \text{SoftMin}_\gamma \left([r_{n-i,t-j,t'-k}]_{(i,j,k)\in\Pi} \right)$

Output: $\text{SoftMin}_\gamma \left([r_{n,\tau,\tau'}]_{n\in\{-\eta,...,\eta\}} \right)$

- Suppose we tackle the camera viewpoint alignment in just camera azimuth.
- The maximum viewpoint change from block to block is ι-max shift (smoothness).

- We initialize all possible origins of shifts in accumulator $r_{n,1,1}$.
- A phase related to soft-DTW (temporal-viewpoint alignment) takes place.
- We choose the path with the smallest distance (of matched features) over all possible viewpoint ends by selecting a soft-minimum over $[r_{n,\tau,\tau'}]_{n\in\{-\eta,...,\eta\}}$.

View-wise Soft-DTW *vs.* FVM *vs.* JEANIE



(a) soft-DTW (view-wise) (b) FVM (c) JEANIE(1-max shift)

Figure 13: The support & query sequence are shown in green & black respectively.

- soft-DTW finds each individual alignment **per viewpoint fixed** throughout alignment: $d_{\text{shortest}} = 4.08$. Too pessimistic!
- FVM is a **greedy matching algorithm** which leads to unrealistic zigzag path: $d_{\text{FVM}} = 2.53$. Overoptimistic!
- JEANIE (1-max shift) is able to find **smooth joint viewpoint-temporal alignment** between support and query sequences: $d_{\text{JEANIE}} = 3.69$.

Free Viewpoint Matching (FVM) **seeks the best local viewpoint alignment** **for every step of DTW**, thus resulting in a **non-smooth path along viewpoint axis** in contrast to JEANIE.

Pipeline: further details

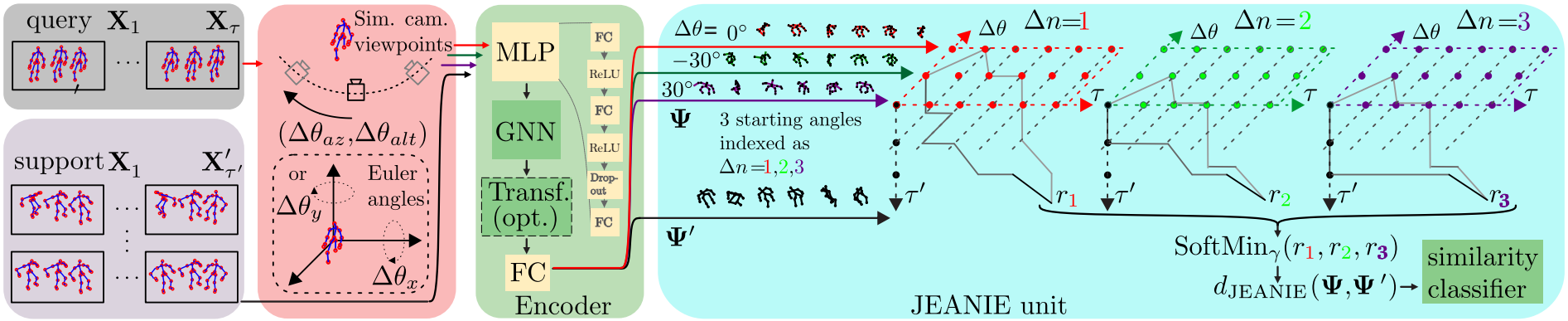
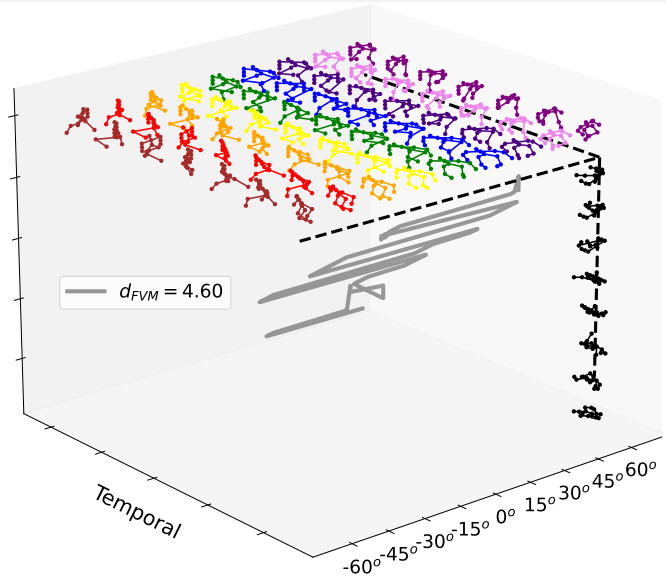


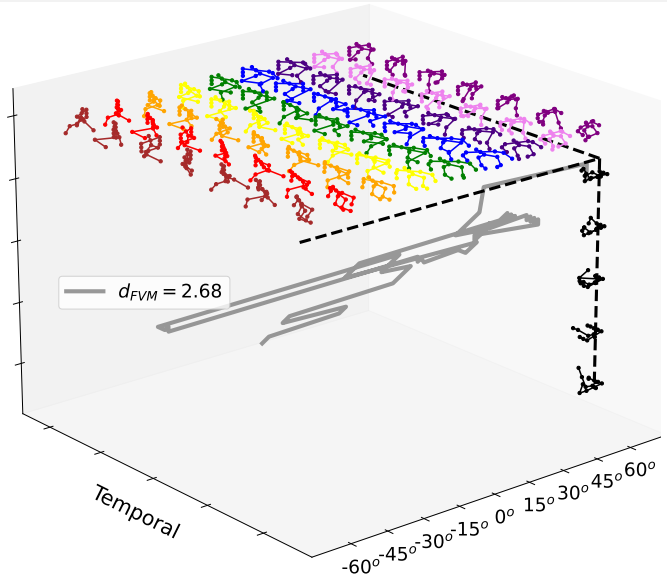
Figure 14: Our 3D skeleton-based FSAR with JEANIE.

- Generate multiple rotations by $(\Delta\theta_x, \Delta\theta_y)$ of each query by
  - **Euler angles** (baseline approach) or
  - **simulated camera views** (gray cameras) by camera shifts $(\Delta\theta_{az}, \Delta\theta_{alt})$.
- Temporal-viewpoint alignment takes place in 4D space (we show a 3D case).
- **Temporally-wise**, JEANIE starts from the same $t=(1,1)$ & finishes at $t=(\tau, \tau')$.
- **Viewpoint-wise**, JEANIE starts from **every possible camera shift** & finishes at one of possible camera shifts.
- At each step, the step may be no larger than $(\pm\Delta\theta_{az}, \pm\Delta\theta_{alt})$ to prevent erroneous alignments.

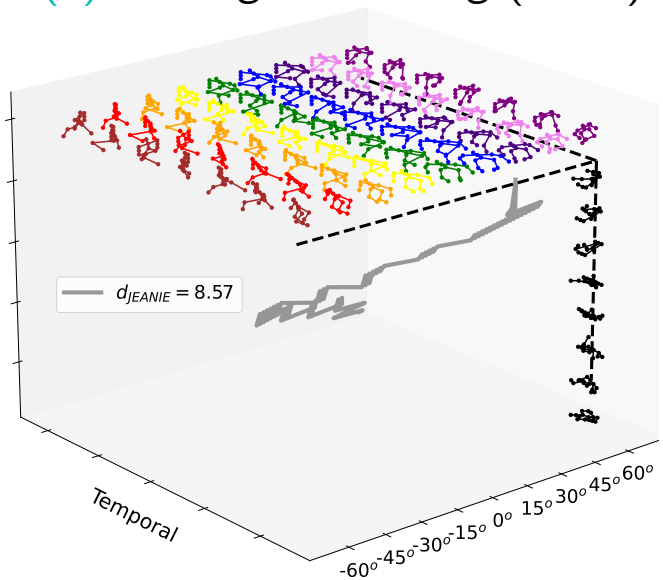
Results & Discussions



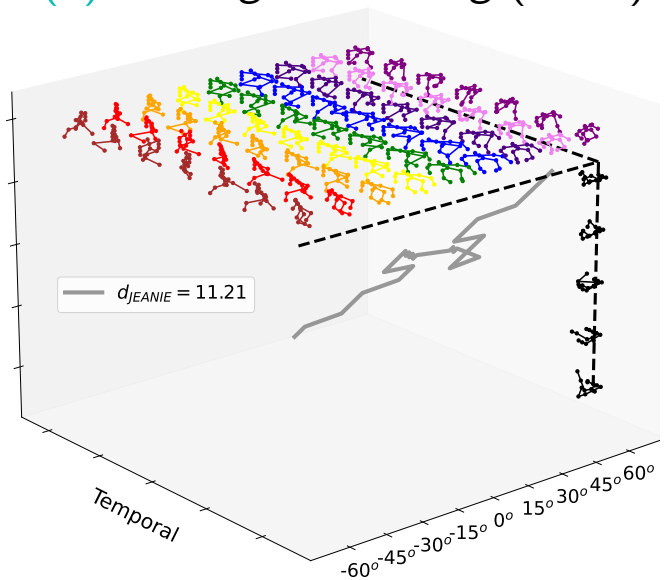
(a) *walking vs. walking* (**FVM**)



(b) *walking vs. running* (**FVM**)



(c) *walking vs. walking* (**JEANIE**)



(d) *walking vs. running* (**JEANIE**)

Results & Discussions (cont.)

Table 7: Results on NTU-120 (multiview classification).

Training view / Testing view	bott. cent.	bott. top	bott.& cent. top	left cent.	left right	left & cent. right
100/same 100 (baseline)	74.2	73.8	75.0	58.3	57.2	68.9
100/same 100 (*FVM*)	79.9	78.2	80.0	65.9	63.9	75.0
100/same 100 (*JEANIE*)	**81.5**	**79.2**	**83.9**	**67.7**	**66.9**	**79.2**
100/novel 20 (baseline)	58.2	58.2	61.3	51.3	47.2	53.7
100/novel 20 (*FVM*)	66.0	65.3	68.2	58.8	53.9	60.1
100/novel 20 (*JEANIE*)	**67.8**	**65.8**	**70.8**	**59.5**	**55.0**	**62.7**

Table 8: Experiments on 2D and 3D Kinetics-skeleton.

	S^2GC (no soft-DTW)	soft-DTW	*FVM*	JEANIE	JEANIE +Transf.
2D skel.	32.8	34.7	-	-	-
3D skel.	35.9	39.6	44.1	**50.3**	**52.5**

Discussion.

- Few-shot multi-view classification.
  - Adding more camera viewpoints helps.
  - Even with (*novel 20*) (not used in training), we still achieve 62.7% & 70.8%.
- JEANIE on the Kinetics-skeleton dataset.
  - We use Euler angles.
  - 3D outperforms 2D by 3–4%.
  - With Transformer, JEANIE further boosts results by 2%.
- For more details, see our paper[9].

[9] Wang, L., & Koniusz, P. (2022). **Temporal-Viewpoint Transportation Plan for Skeletal Few-shot Action Recognition**. In *ACCV*, oral, **Best Student Paper Award**.

Motivation

- GCN-based
  - represent human body joints based on **physical connectivity**
  - **limited** receptive fields / one- or few-hop neighbourhood aggregation
  - ignore the dependency between body joints **non-connected** by body parts
- Human actions are associated with **interaction groups of skeletal joints**
  - the impact of groups of joints on each action differs
- Inspired by our tensor representations[10]:
  - *sequence compatibility kernel* (SCK) & *dynamics compatibility kernel* (DCK)
  - compactly **capture complex interplay**
  - operate on **subsequences** / capture the local-global interplay of correlations
  - incorporate **multi-modal inputs**

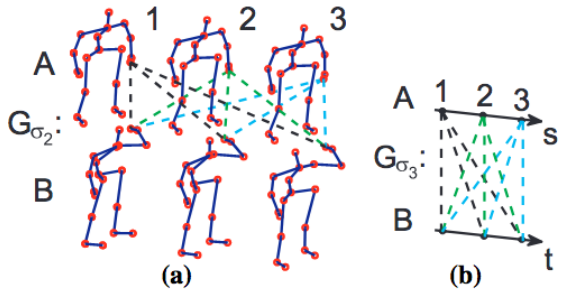


Figure 16: SCK

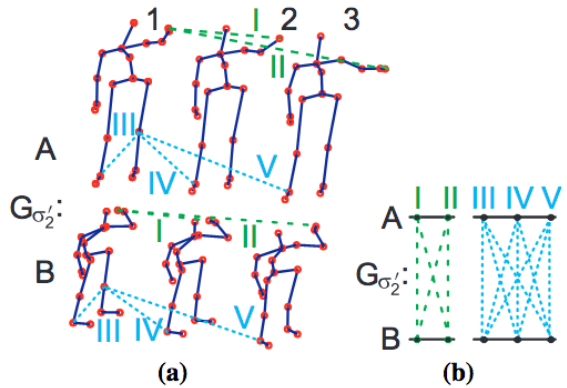


Figure 17: DCK

[10] Koniusz, P., Wang, L., & Cherian, A. (2021). **Tensor representations for action recognition**. *IEEE TPAMI*, 44(2), 648-665.

Motivation (cont.)

We propose to:

- use skeletal hypergraph
- Hypergraph captures higher-order relationships by hyper-edges
- Hyper-edges connect more than two nodes (body joints)

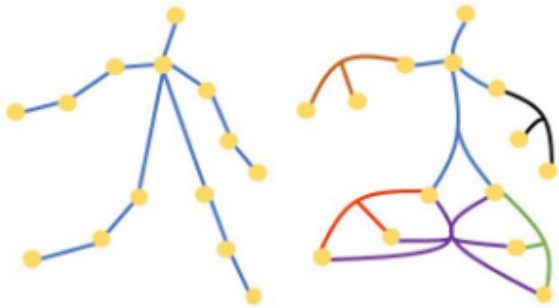


Figure 18: Skeletal graph & hypergraph.

Compared to GCN:

- encodes **first-**/**second-**/ **higher-**order hyper-edges
- set of body joints (**nodes**)/ **edges** between pairs of nodes/**hyper-edges** between triplets of nodes

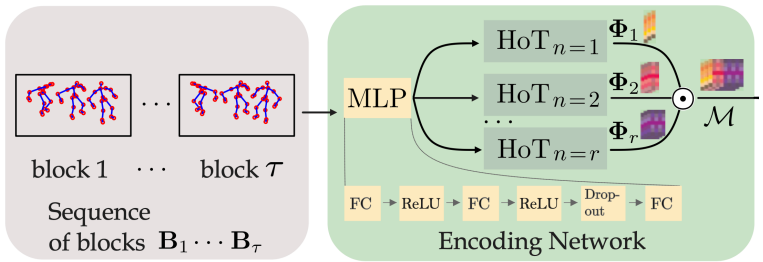


Figure 19: MLP+HoT branches

Concatenating HoT outputs of orders 1 to r across τ[11] blocks is *sub-optimal*.

- #hyper-edges of J joints **grows rapidly with order** r, *i.e.*, $\binom{J}{i}$ for $i = 1, ..., r$
- embeddings of the **highest order hyper-edges dominate lower orders**
- **long-range temporal dependencies** of features are insufficiently explored

[11] For brevity, we write that we have τ temporal blocks per sequence. In fact, τ varies.

Multi-order Multi-mode Transformer (3Mformer)

Given $\mathcal{M} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_r}$, we perform mode-$m$ matricization to obtain $\mathbf{M} \equiv \mathcal{M}_{(m)}^{\top} \in \mathbb{R}^{(I_1 \ldots I_{m-1} I_{m+1} \ldots I_r) \times I_m}$ to form joint-token.

- **joint-mode tokens**:
  - 'channel-temporal block' (Attention matrix $\mathbf{A}_{\text{MP}} \in \mathbb{R}^{d'\tau \times d'\tau}$)
  - 'channel-body joint' ($\mathbf{A}_{\text{TP}} \in \mathbb{R}^{rd'J \times rd'J}$)
  - 'channel-hyper-edge (any order)' ($\mathbf{A}_{\text{TP}} \in \mathbb{R}^{d'N \times d'N}$ & $N = \sum_{m=1}^{r} \binom{J}{m}$)
  - and 'channel-only' ($\mathbf{A}_{\text{MP}} \in \mathbb{R}^{d' \times d'}$) pairs
- **Joint-mode Self-Attention (JmSA)**:
  - show diagonal / vertical patterns
  - patterns are consistent with the pattens of attention matrices found in standard Transformer, *e.g.*, NLP
  - joint-mode attention captures richer information

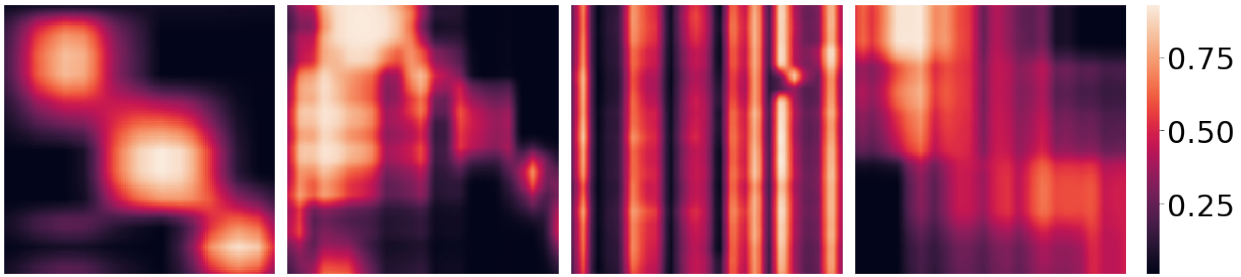


Figure 20: Visualization of attention matrices: 'channel-only', 'channel-hyper-edge', 'order-channel-body joint' & 'channel-temporal block' tokens.

Visualization of 3Mformer

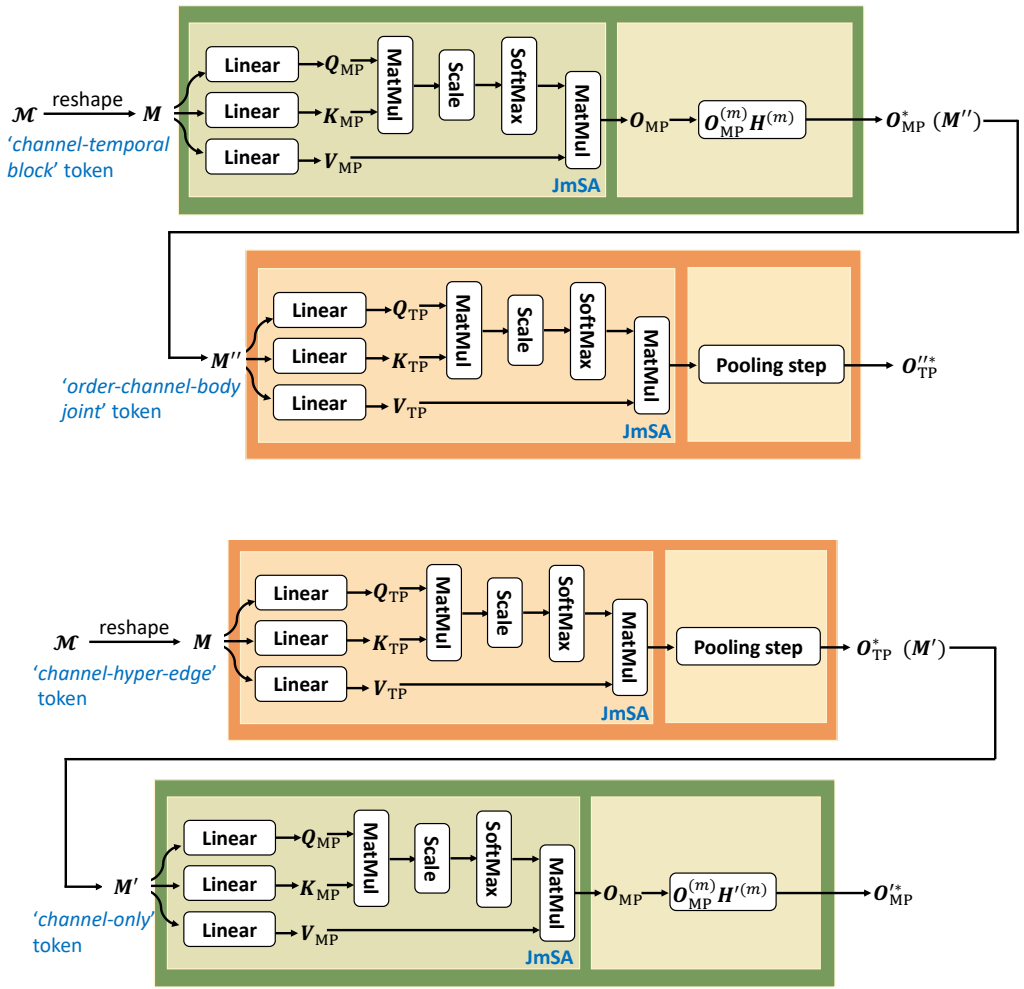


Figure 21: 3Mformer is a two-branch model: (a) MP→TP & (b) TP→MP.

Two basic building modules:

- Multi-order Pooling (MP)
  - combine information flow **block-wise**
  - **various joint-mode** tokens help improve results
  - **different focus** of each attention mechanism
- Temporal block Pooling (TP)
  - each sequence may contains a different number of blocks
  - aggregates via popular pooling, *e.g.*, rank-, first-, second- or higher-order pooling

We also form our **multi-head** JmSA as in standard Transformer.

Pipeline: further details

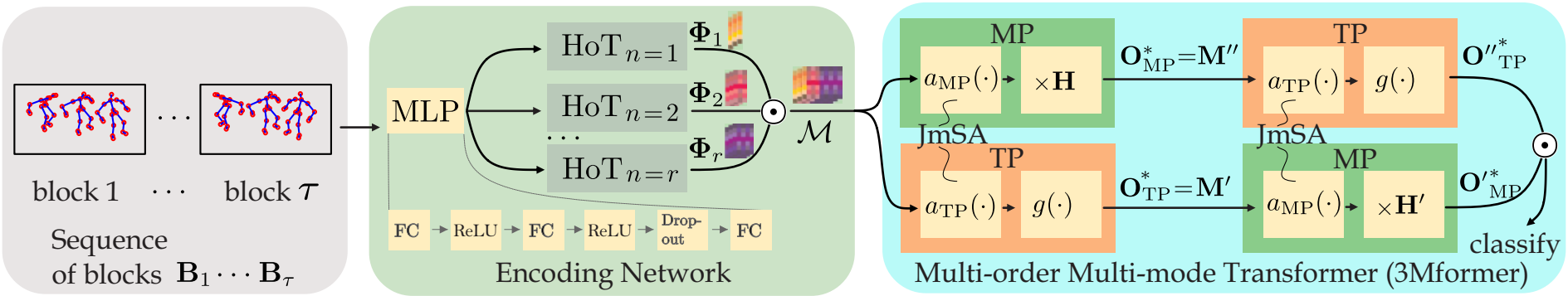


Figure 22: Pipeline overview.

- each sequence is split into τ temporal blocks $\mathbf{B}_1, ..., \mathbf{B}_\tau$
- each block is embedded by a simple MLP into $\mathbf{X}_1, ..., \mathbf{X}_\tau$
- $\mathbf{X}_1, ..., \mathbf{X}_\tau$ are passed to HoTs $(n=1, ..., r)$ for feature tensors $\mathbf{\Phi}_1, ..., \mathbf{\Phi}_\tau$
- subsequently concatenated by $\odot$ along the hyper-edge mode into tensor $\mathbf{M}$
- **3Mformer contains two complementary branches**: MP→TP & TP→MP
- outputs are concatenated by $\odot$ and passed to the classifier
- MP & TP perform attention with the so-called **joint-mode tokens**
- MP contains **weighted pooling along hyper-edge mode** by learnable matrix $\mathbf{H}$ (and $\mathbf{H}'$ in another branch).
- TP contains **block-temporal pooling** denoted by $g(\cdot)$ to capture block-temporal order with pooling

Results & Discussions

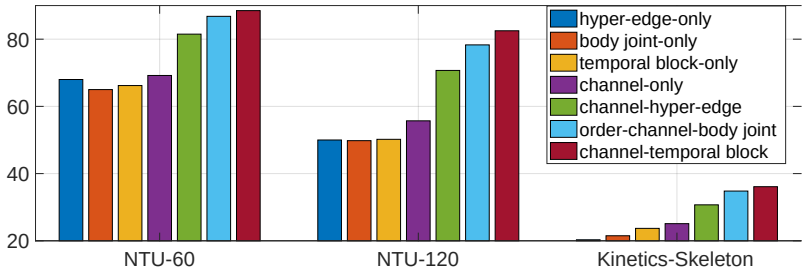


Figure 23: Single-mode tokens *vs.* joint-mode tokens.

Table 9: NTU-60, NTU-120 & Kinetics-Skeleton.

	Method	Venue	NTU-60		NTU-120		Kinetics-Skeleton	
			X-Sub	X-View	X-Sub	X-Set	Top-1	Top-5
Graph-based	TCN	CVPRW'17	-	-	-	-	20.3	40.0
	ST-GCN	AAAI'18	81.5	88.3	70.7	73.2	30.7	52.8
	AS-GCN	CVPR'19	86.8	94.2	78.3	79.8	34.8	56.5
	2S-AGCN	CVPR'19	88.5	95.1	82.5	84.2	36.1	58.7
	NAS-GCN	AAAI'20	89.4	95.7	-	-	37.1	60.1
	Sym-GNN	TPAMI'22	90.1	96.4	-	-	37.2	58.1
	Shift-GCN	CVPR'20	90.7	96.5	85.9	87.6	-	-
	MS-G3D	CVPR'20	91.5	96.2	86.9	88.4	38.0	60.9
Hypergraph-based	Hyper-GNN	TIP'21	89.5	95.7	-	-	37.1	60.0
	DHGCN	CoRR'21	90.7	96.0	86.0	87.9	37.7	60.6
	Selective-HCN	ICMR'22	90.8	96.6	-	-	38.0	61.1
	SD-HGCN	ICONIP'21	90.9	96.7	87.0	88.2	37.4	60.5
Transformer-based	ST-TR	CVIU'21	90.3	96.3	85.1	87.1	38.0	60.5
	MTT	LSP'21	90.8	96.7	86.1	87.6	37.9	61.3
	4s-GSTN	Symmetry'22	91.3	96.6	86.4	88.7	-	-
	STST	ACM MM'21	91.9	96.8	-	-	38.3	61.2
	3Mformer (with avg-pool, *ours*)		92.0	97.3	88.0	90.1	43.1	65.2
	3Mformer (with max-pool, *ours*)		92.1	97.8	-	-	-	-
	3Mformer (with attn-pool, *ours*)		**94.2**	**98.5**	89.7	92.4	45.7	67.6
	3Mformer (with tri-pool, *ours*)		94.0	**98.5**	**91.2**	**92.7**	**47.7**	**71.9**
	3Mformer (with rank-pool, *ours*)		**94.8**	**98.7**	**92.0**	**93.8**	**48.3**	**72.3**

Discussions:

- Single-mode tokens *vs.* joint-mode tokens
- graph-based *vs.ours*:
  - AS-GCN/2S-AGCN
    - pairwise relationship
    - second-order
  - *ours*
    - higher-order
    - groups of body joints
  - 2nd-order HoT alone *vs.* NAS-GCN/Sym-GNN
- hypergraph-based *vs.ours*:
  - 3rd-order HoT alone *vs.* Hyper-GNN/SD-HGCN/Selective-HCN

Conclusion & Future Work

Conclusion & Future Work

Conclusion:

- Video-based:
  - self-supervision/hallucination-based
  - + easier to obtain video frames/rich visual information/robust backbones
  - – require large-scale dataset/computational cost
- Skeleton-based:
  - tensor representations & 3Mformer
  - + openpose & Kinect toolkit+OpenNI/lightweight/faster to process
  - – require large-scale dataset/reliability/ lack visual information
- Few-shot:
  - alignment-based/match query-support pair
  - + faster adaptation to novel classes/limited data is fine
  - – robust data is required to learn a good model

Future work:

- extending current models to **anomaly detection**
- **balancing** short-term temporal and long-term patterns
- smarter ways of getting **reliable motion description** (similar to 3D body joints but more flexible) from RGB-D video

Thank you!