요약

https://github.com/yysijie/st-gcn

- Skeleton-base의 모션 데이터를 그래프로 표현하여 Graph Convolution Network(GCN)를 사용한 대표 사례 - 약 1500회 인용
    - 타 skeleton-base 모델보다 성능이 좋고 RGB based 모델보다는 성능이 떨어짐
- skeleton hierarchy 구조로 Graph를 표현(본은 node, 연결 여부를 edge)
  - spatial 영역
- 시간적 연결성을 표현하기 위해서 같은 본의 다음 프레임/이전 프레임과도 연결한 그래프로 하나의 모션파일 표현
  - temporal 영역
- Convolution에 사용할 주변 노드 선택(+라벨링) 방법론을 제안
- Convolution 특성상 멀리 떨어진 노드(본)간 관계를 학습하지못하는 한계가 있음
    - 후속논문에서 개선시킴

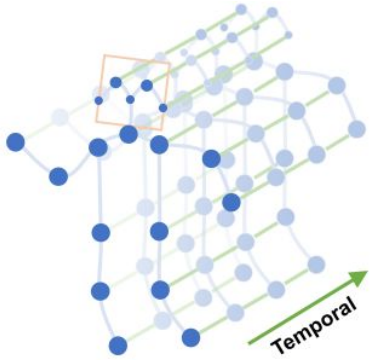
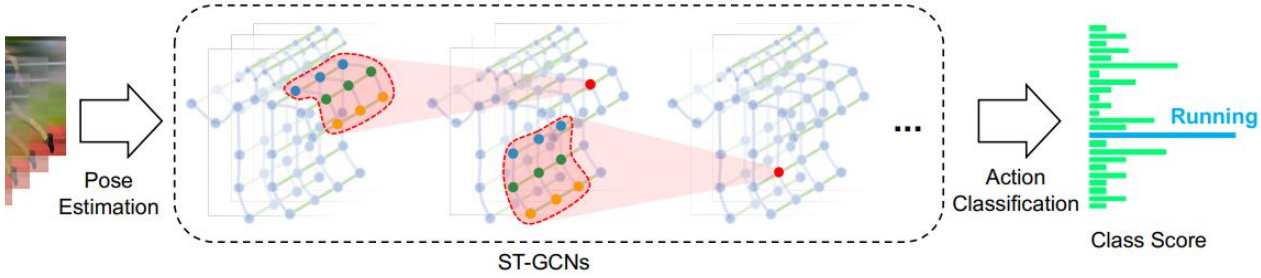


Figure 1: The spatial temporal graph of a skeleton sequence used in this work where the proposed ST-GCN operate on. Blue dots denote the body joints. The intra-body edges between body joints are defined based on the natural connections in human bodies. The inter-frame edges connect the same joints between consecutive frames. Joint coordinates are used as inputs to the ST-GCN.

ST-GCN



- 모션을 그래프 표현한 뒤 graph convolution network 제안
- 단일 프레임 기준 CNN 수식(1) => S-GCN 식(5)로 변환 (Spatial-Graph Convolution Network)
  - 샘플링 : 선택 픽셀 주변의 커널 사이즈 만큼의 픽셀(CNN) => partition strategies 기반(3페이지)의 선택 노드 주변 선택
    - partition strategies 으로 선택된 주변 노드는 label을 붙여서 사용할 weight가 결정됨
  - 가중치 : 선택 픽셀 기준 위치에 따른 weight 사용(CNN) => 지정된 label에 따른 weight 사용
- 다중 프레임 기준 S-GCN 수식(5)에 노드 선택 및 라벨링 수식 변경 [식- (6),(7)]
  - 단일 프레임에서 partition strategies으로 선택된 노드의 과거/미래 프레임에 대해서 추가 샘플링
- 식 1) CNN 수식 kernel size K일때, sampling func p와 weight function w로 주변 height h, width w 만큼을 샘플링하고 가중치 적용
- 식 2) 본 논문이 제안한 partition strategies(distance func d)로 인접 노드 샘플링 v_ti : 시간 t에서 선택된 i번째 노드
- 식 3) 인접 노드에 붙은 label에 따라 가중치 선택 w의 shape (라벨 사이즈 K, 차원 수 C)
- 식 4, 5) spatial graph convolution network 표현 normalize func Z가 추가
- 식 6) 식 2에서 공간적 인접 노드에 파라미터 T/2 만큼 과거/미래 프레임에 연결된 노드를 시간적 인접성 부여
- 식 7) 시간적 인접 노드의 label적용 방법

$$f_{out}(\mathbf{x}) = \sum_{h=1}^{K}\sum_{w=1}^{K} f_{in}(\mathbf{p}(\mathbf{x},h,w)) \cdot \mathbf{w}(h,w), \quad (1)$$

$$B(v_{ti}) = \{v_{tj} | d(v_{tj}, v_{ti}) \leq D\} \; \mathbf{p}(v_{ti}, v_{tj}) = v_{tj}. \quad (2)$$

$$l_{ti} : B(v_{ti}) \rightarrow \{0, \ldots, K-1\} \; \mathbf{w}(v_{ti}, v_{tj}) = \mathbf{w}'(l_{ti}(v_{tj})). \quad (3)$$

$$f_{out}(v_{ti}) = \sum_{v_{tj} \in B(v_{ti})} \frac{1}{Z_{ti}(v_{tj})} f_{in}(\mathbf{p}(v_{ti}, v_{tj})) \cdot \mathbf{w}(v_{ti}, v_{tj}), \quad (4)$$

$$f_{out}(v_{ti}) = \sum_{v_{tj} \in B(v_{ti})} \frac{1}{Z_{ti}(v_{tj})} f_{in}(v_{tj}) \cdot \mathbf{w}(l_{ti}(v_{tj})). \quad (5)$$

$$B(v_{ti}) = \{v_{qj} | d(v_{tj}, v_{ti}) \leq K, |q-t| \leq \lfloor \Gamma/2 \rfloor\}. \quad (6)$$

$$l_{ST}(v_{qj}) = l_{ti}(v_{tj}) + (q - t + \lfloor \Gamma/2 \rfloor) \times K, \quad (7)$$

Partition Strategies

- 논문은 S-GCN 단계에서 인접노드 선택 방법들을 제안
  - Uni-labeling (b)
    - 선택노드 + 연결된 인접노드 다 동일 label 부여
    - weight w 의 사이즈를 (label 수 K=1, 차원수 C)로 해서 동일한 가중치를 적용하겠다
  - Distance partitioning (c)
    - 선택 노드와 멀어질수록 label값을 1씩 늘리자
    - Distance D = 1 인경우, 선택된 node는 label 0, 인접한 노드는 label 1로 통일
    - root node와 인접노드간 가중치를 따로 적용하겠다
  - Spatial configuration partitioning (d) - 식 (8)
    - root node와 가까운 인접노드 / 선택된 노드 / 하위 노드를 구분해서 label을 설정하자

$$l_{ti}(v_tj) = \begin{cases} 0 & \text{if } r_j = r_i \\ 1 & \text{if } r_j < r_i \\ 2 & \text{if } r_j > r_i \end{cases} \quad (8)$$

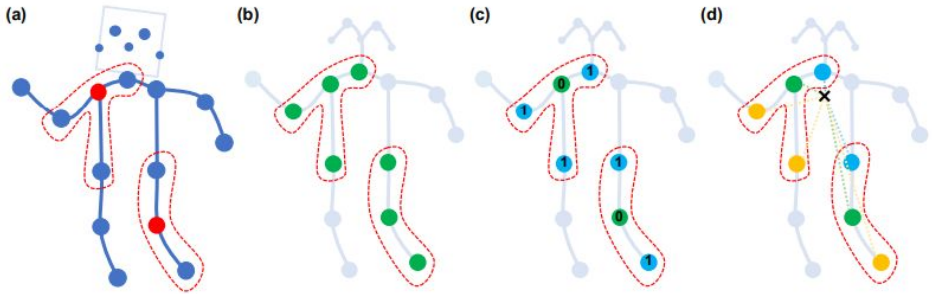


Figure 3: The proposed partitioning strategies for constructing convolution operations. From left to right: **(a)** An example frame of input skeleton. Body joints are drawn with blue dots. The receptive fields of a filter with $D = 1$ are drawn with red dashed circles. **(b) Uni-labeling** partitioning strategy, where all nodes in a neighborhood has the same label (green). **(c) Distance** partitioning. The two subsets are the root node itself with distance 0 (green) and other neighboring points with distance 1. (blue). **(d) Spatial configuration** partitioning. The nodes are labeled according to their distances to the skeleton gravity center (black cross) compared with that of the root node (green). Centripetal nodes have shorter distances (blue), while centrifugal nodes have longer distances (yellow) than the root node.

실험

- dataset : kinect, NTU-RGB+D
  - kinect dataset은 skeleton 정보가 없어서 openpose로 저자가 직접 만듦
- partitioning strategies 비교 (표 1)
  - Baseline TCN : 본 별 모든 프레임을 concat해서 연산
  - Local Convolution : Graph로 데이터를 표현했지만 FC를 사용
  - Distance partitioning* : Uni-labeling + Distance Partitioning 같이 사용
  - ST-GCN + Imp. : Distance Partitioning + Spatial Configuration 같이 사용
- 다른 모델과 성능 비교 (표 2-4)
  - 표2) 같은 데이터를 다른 포맷으로 만들어서 사용한 기법들과 비교
  - 표3) Skeleton based model 끼리 비교
  - 표4) 표2에서 사용한 Kinectics dataset에서 action 수를 줄여서 다시 실험
  - 표5) 다른 데이터 포맷과 앙상블도 해봄

	Top-1	Top-5
Baseline TCN	20.3%	40.0%
Local Convolution	22.0%	43.2%
Uni-labeling	19.3%	37.4%
Distance partitioning*	23.9%	44.9%
Distance Partitioning	29.1%	51.3%
Spatial Configuration	29.9%	52.2%
ST-GCN + Imp.	**30.7%**	**52.8%**

Table 1: Ablation study on the Kinetics dataset. The "ST-GCN+Imp." is used in comparison with other state-of-the-art methods. For meaning of each setting please refer to Sec.4.2.

Method	RGB CNN	Flow CNN	ST-GCN
Accuracy	70.4%	72.8%	72.4%

Table 4: Mean class accuracies on the "Kinetics Motion" subset of the Kinetics dataset. This subset contains 30 action classes in Kinetics which are strongly related to body motions.

	Top-1	Top-5
RGB(Kay et al. 2017)	57.0%	77.3%
Optical Flow (Kay et al. 2017)	49.5%	71.9%
Feature Enc. (Fernando et al. 2015)	14.9%	25.8%
Deep LSTM (Shahroudy et al. 2016)	16.4%	35.3%
Temporal Conv. (Kim and Reiter 2017)	20.3%	40.0%
ST-GCN	**30.7%**	**52.8%**

Table 2: Action recognition performance for skeleton based models on the Kinetics dataset. On top of the table we list the performance of frame based methods.

	X-Sub	X-View
Lie Group (Veeriah, Zhuang, and Qi 2015)	50.1%	52.8%
H-RNN (Du, Wang, and Wang 2015)	59.1%	64.0%
Deep LSTM (Shahroudy et al. 2016)	60.7%	67.3%
PA-LSTM (Shahroudy et al. 2016)	62.9%	70.3%
ST-LSTM+TS (Liu et al. 2016)	69.2%	77.7%
Temporal Conv (Kim and Reiter 2017).	74.3%	83.1%
C-CNN + MTLN (Ke et al. 2017)	79.6%	84.8%
ST-GCN	**81.5%**	**88.3%**

Table 3: Skeleton based action recognition performance on NTU-RGB+D datasets. We report the accuracies on both the cross-subject (X-Sub) and cross-view (X-View) benchmarks.

	RGB TSN	Flow TSN	ST-GCN	Acc(%)
Single Model	✓			70.3
		✓		51.0
			✓	30.7
Ensemble Model	✓	✓		71.1
	✓		✓	71.2
	✓	✓	✓	71.7

Table 5: Class accuracies on the Kinects dataset **without** ImageNet pretraining. Although our skeleton based model ST-GCN can not achieve the accuracy of the state of the art model performed on RGB and optical flow modalities, it can provide stronger complementary information than optical flow based model.