Recognizing Human Actions as the Evolution of Pose Estimation Maps

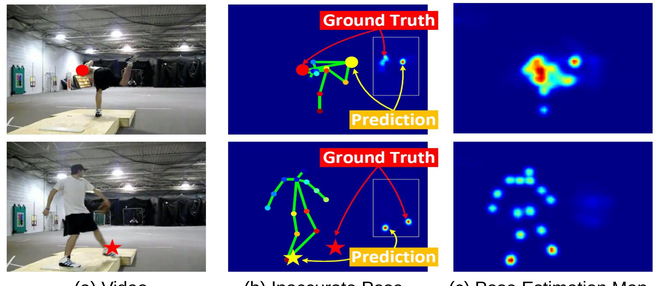
Mengyuan Liu[1], Junsong Yuan[2]

Nanyang Technological University[1], University at Buffalo[2]

liumengyuan@ntu.edu.sg jsyuan@buffalo.edu

Motivation

- Human action recognition using 3d poses estimated from depth sensor has achieved success with deep learning methods.
- Estimated 2d poses from RGB sensor are usually noisy due to partial occlusions and self-similarities.
- Can we recognize human actions from these noisy 2d poses?

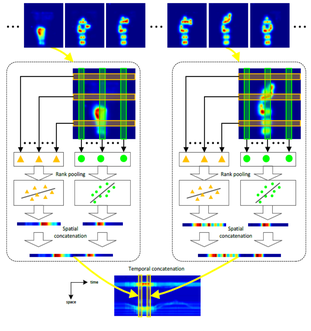


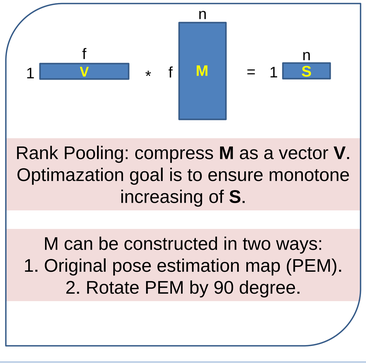
(a) Video (b) Inaccurate Pose (c) Pose Estimation Map

Fig 1: Pose estimation map (PEM, accumulation of joint estimation maps) provides global body shape, which improves noisy 2d poses on recognition task.

Signature from Evolution of PEMs (S-PEM)

- Use Rank Pooling to compress each frame as a vertical vector.
- Concatenate vectors as an image, according to their temporal order.





Rank Pooling: compress **M** as a vector **V**. Optimazation goal is to ensure monotone increasing of **S**.

M can be constructed in two ways:
1. Original pose estimation map (PEM).
2. Rotate PEM by 90 degree.

Fig 3: Generation of S-PEM

Overview of the Proposed Method

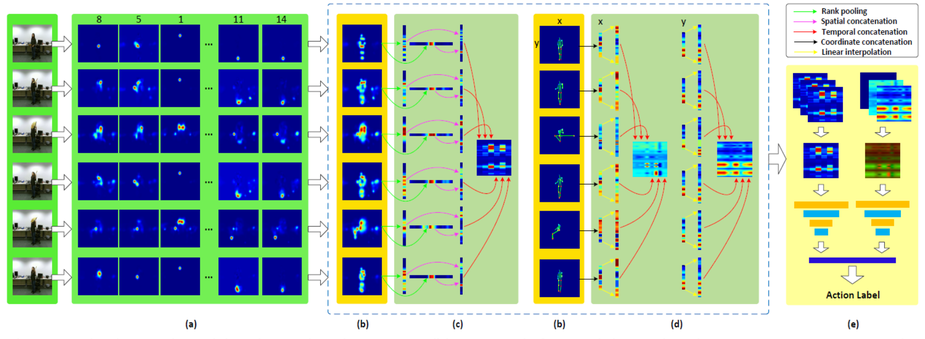


Fig 2: Method overview. (a) Joint estimation maps. (b) For each frame, joint estimation maps are aggregated to form a pose estimation map (PEM) and a pose. (c) The sequence of PEMs is described as an action signature called S-PEM. (d) The sequence of poses is described as an action signature called S-P. (e) Late fusion.

Signature from Evolution of Poses (S-P)

- Describe each pose as two vertical vectors, i.e., X = (x_1, …, x_{14}) and Y = (y_1, …, y_{14}). The joint order is rearranged (see right figure).
- Concatenate vector X (or Y) as an image, according to their temporal order.
- Obtain an image with two channels.

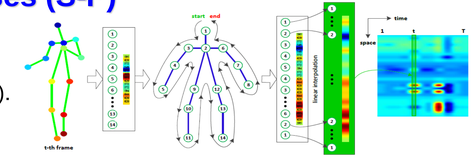
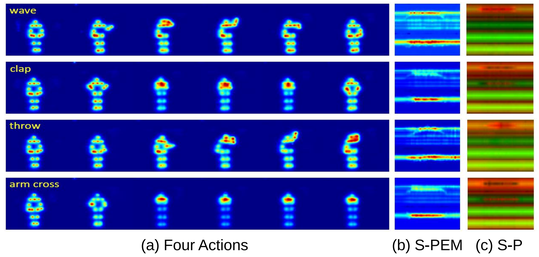


Fig 4: Generation of S-P

Two Stream Fusion

- The one channel image (S-PEM) is repeated thee times.
- The two channel image (S-P) is padded with a zero-value channel.
- Images are resized to [224 224].
- Pretrained VGG-19 model is used (Last layer: from 1000 to the number of total actions in dataset).
- Late fusion.



(a) Four Actions (b) S-PEM (c) S-P

Fig 5: Comparision between S-PEM and S-P

Experiments

- NTU RGB+D dataset: the largest one for 3d pose-based recognition task.
- 56880 videos; 60 actions; 40 subjects; various views.
- Cross Subject (CS): 40320 videos for training; 16560 videos for testing.
- Cross View (CV): 37920 videos for training; 18960 videos for testing.

Data	Method	Type	Year	CS	CV
3D Pose (Kinect)	Super Normal Vector	Hand-crafted	2014	31.82%	13.61%
	Deep RNN	RNN	2016	59.29%	64.09%
	GCA-LSTM	Improved RNN	2017	74.40%	82.80%
	Clips + CNN + MTLN	CNN	2017	79.57%	84.83%
S-PEM (RGB)	S-PEM	CNN	-	72.75%	78.35%
S-P (RGB)	S-P	CNN	-	72.96%	77.21%
S-P + S-PEM	Two Stream	CNN	-	78.80%	84.21%
3D Pose (Kinect)	Extended S-P	CNN	-	82.38%	85.75%
3D Pose + S-PEM	Two Stream	CNN	-	91.71%	95.26%

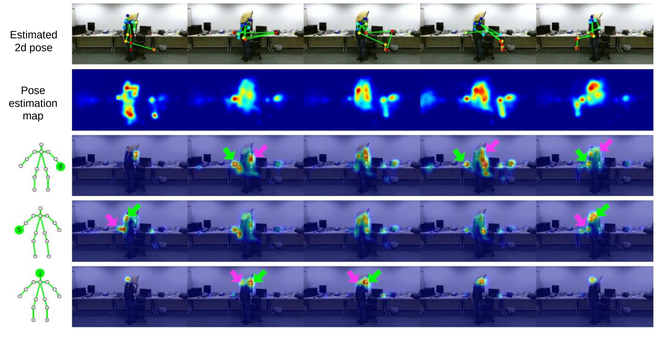


Fig 6: Green arrow points out the estimated position of joint, which is inaccurate. Meanwhile, pink arrow points to the region of PEM which covers the ground truth of the joint position.

Conclusions

- Sole 2d poses from RGB sensor (S-P) performs poorly for recognition task.
- Pose estimation maps (S-PEM) improve performances of 2d poses (S-P).
- The fused form of PEMs and poses (S-P+S-PEM, using RGB) achieves compatable performances with 3d pose-based methods (using Kinect).
- 3D Pose + S-PEM (using both RGB and depth data) performs the best.