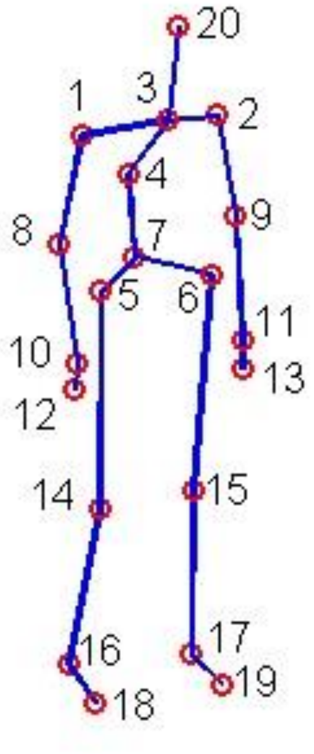
Action Recognition Using Delay Embedding

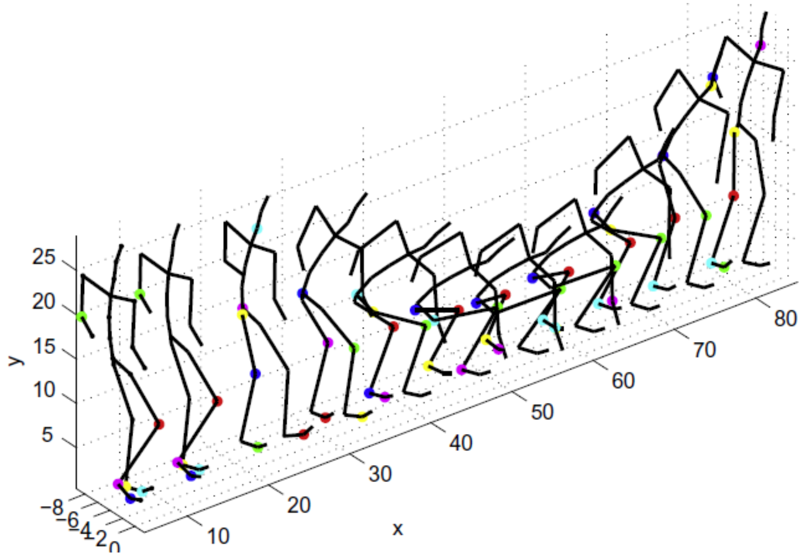
Zhifei Zhang

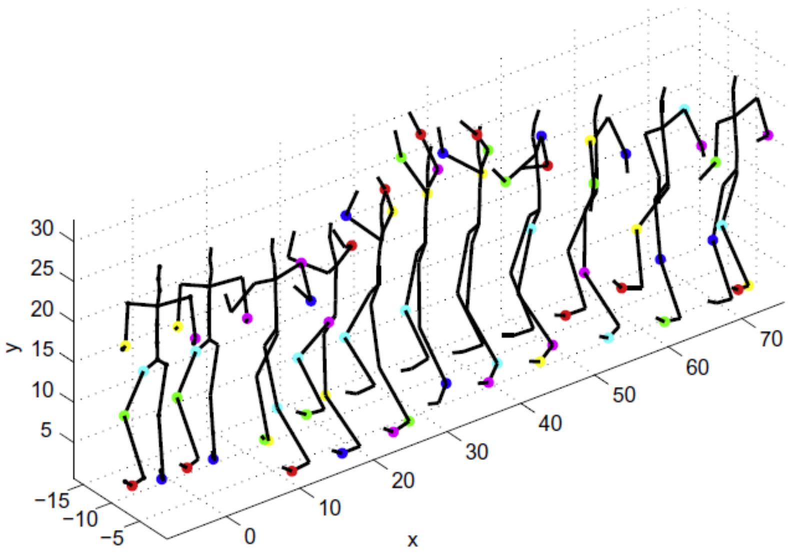
Outline

1. Skeleton-based action recognition
2. Delay embedding on motion
3. Experiments

Skeleton-based action recognition





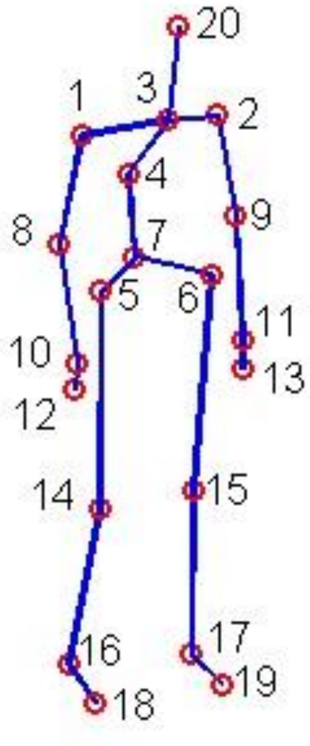


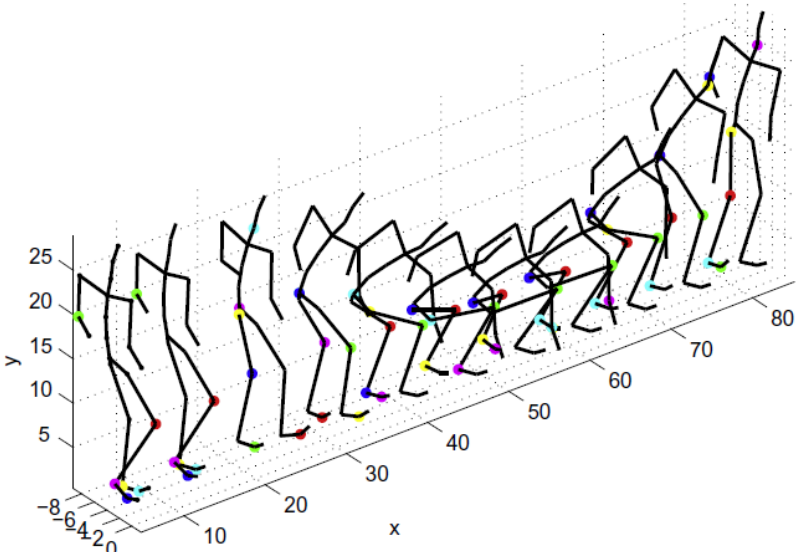
Full Skeleton
or
Body Parts

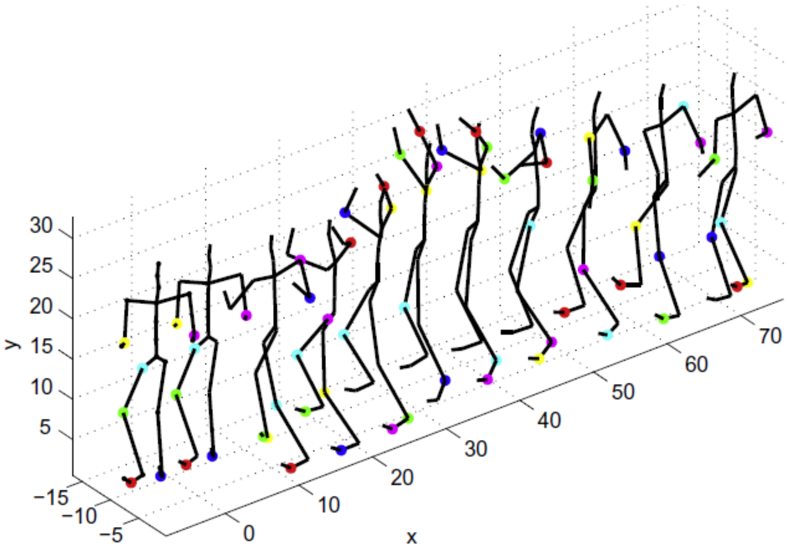
Position
or
Angle

Absolute
or
Relative

Skeleton-based action recognition





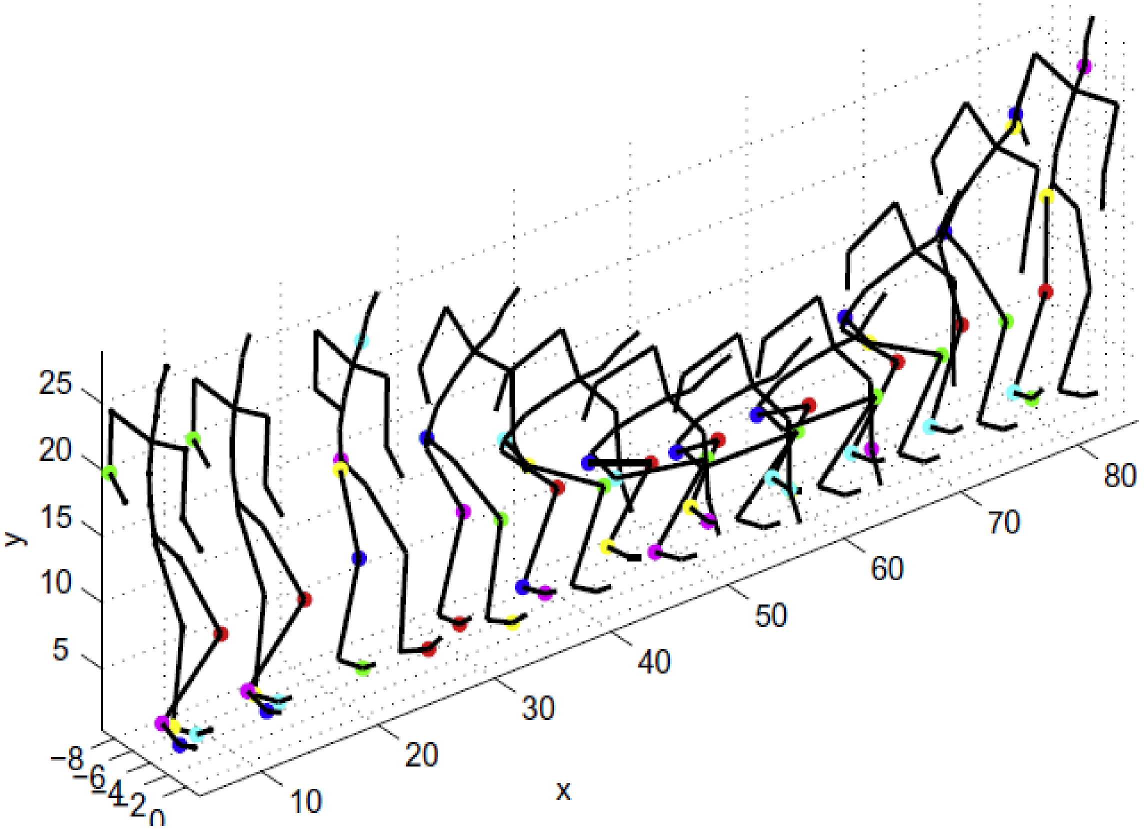


Full Skeleton or **Body Parts**

Position or **Angle**

Absolute or **Relative**

Delay embedding on motion



If focus on the ith joint, we get a 3D time series.

$$J_i = \begin{pmatrix} x_{i1} & & x_{in} \\ y_{i1} & \cdots & y_{in} \\ z_{i1} & & z_{in} \end{pmatrix}^T_{3\times n}$$

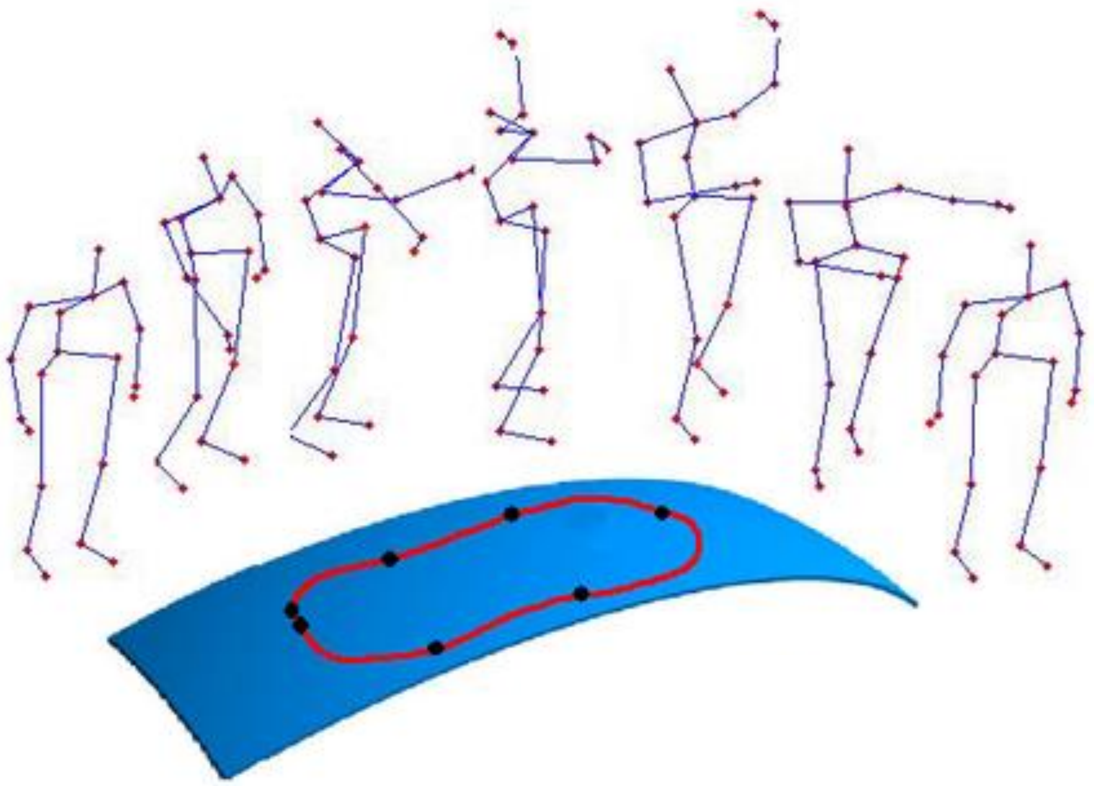
Assume there are n frames and 20 joints.

$$\mathbb{J} = [J_1 \quad \cdots \quad J_{20}]^T_{(3\times 20)\times n}$$

Which is a 60-D time series.

Delay embedding on motion

Delay embedding on a 60-D time series → 120-D curve in the embedding space

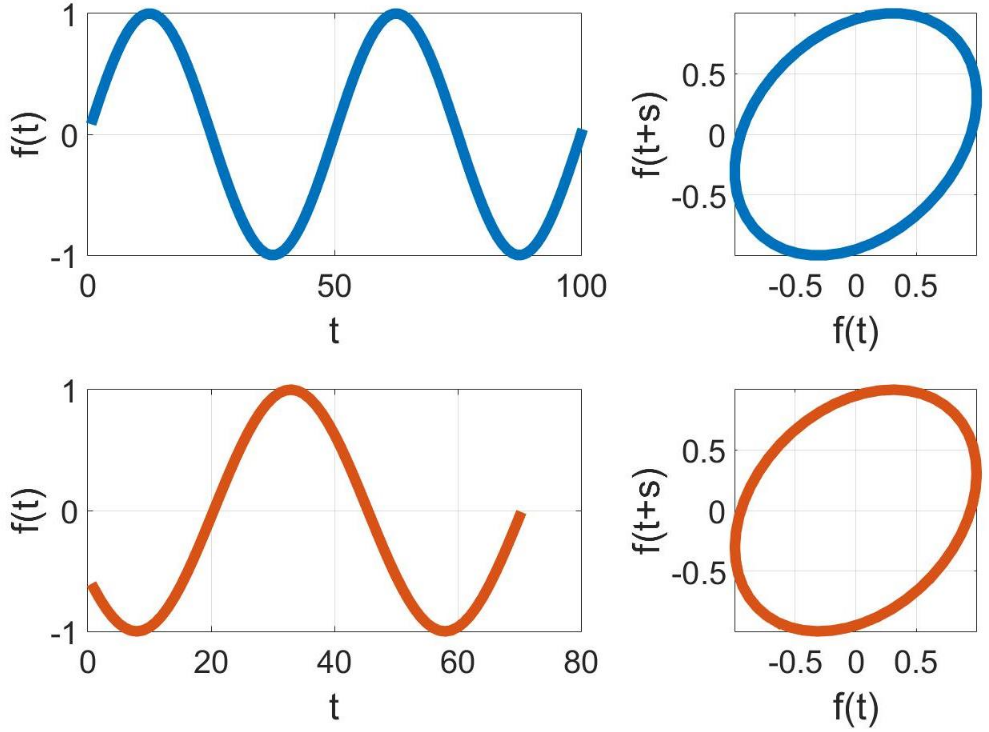


Issues need to concern:

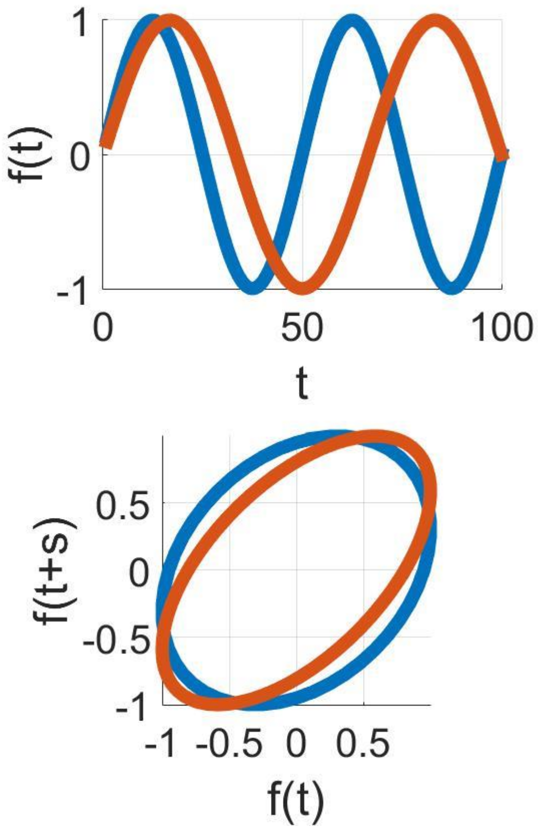
- **Rotation**
- **Translation**
- **Scaling**
- **Temporal misalignment**
- **Rate variation**
- **Deformation**

Delay embedding on motion

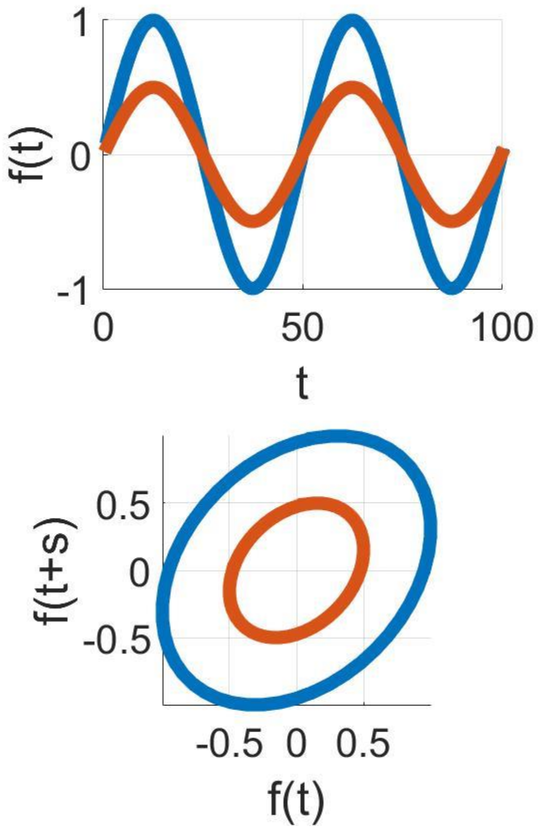
Temporal misalignment



Rate variation

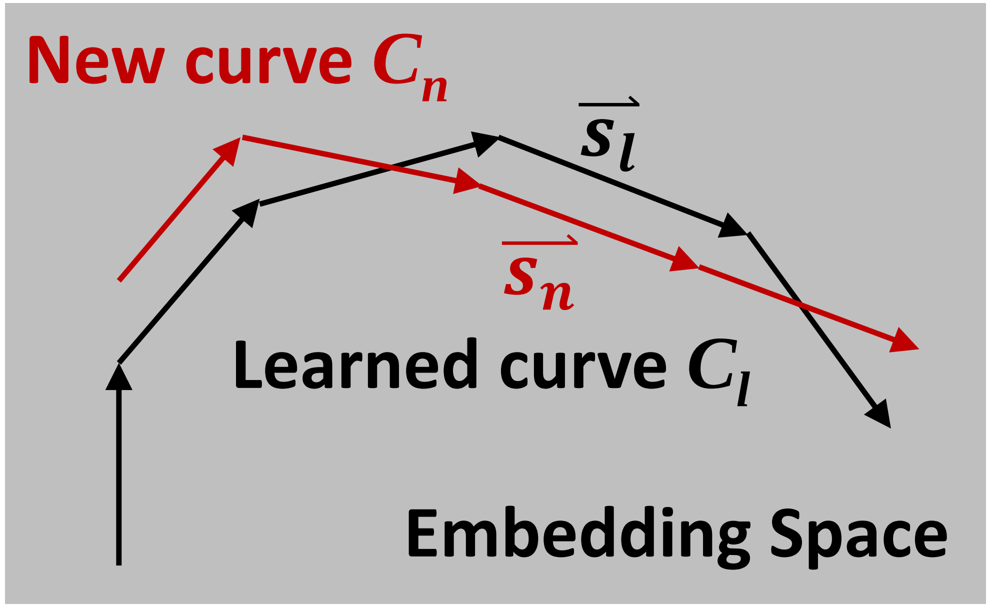


Deformation



Classification --- Trajectory Matching

- **Dynamic Time Warping**
- **Nearest Neighbor**
- **Shape Analysis (SRVF, geodesic)**



$$Dist(C_n, C_l) = \sum_{\vec{s_n} \in C_n} D_{C_l}(\vec{s_n})$$

$$D_C(\vec{s_n}) = \min_C D(\vec{s_n}, \vec{s_l})$$
$$s.t. \quad \vec{s_l} \in C$$

where

$$D(\vec{s_n}, \vec{s_l}) = \underbrace{\frac{\|s_n - s_l\|}{\|\vec{s_n}\|}}_{\text{Euclidian distance}} + \alpha \underbrace{\exp\left(\frac{\|\vec{s_n}\| - \|\vec{s_l}\|}{\|\vec{s_n}\|}\right)}_{\text{Difference of length}} + \beta \underbrace{\exp\left(\arccos \frac{\vec{s_n} \cdot \vec{s_l}}{\|\vec{s_n}\|\|\vec{s_l}\|}\right)}_{\text{Difference of angle}}$$

Classification --- Trajectory Matching

- **Shape Analysis (SRVF, geodesic)**

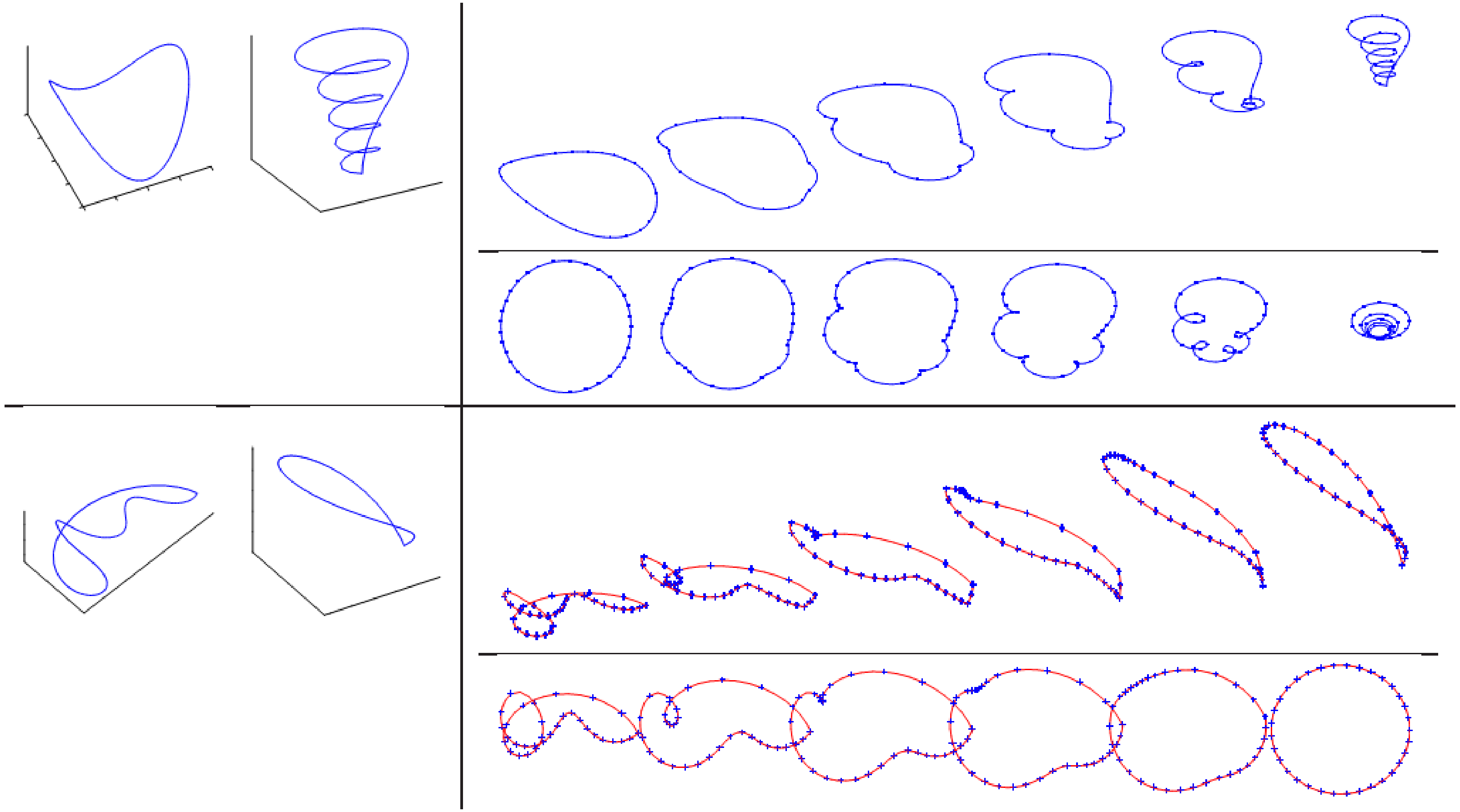
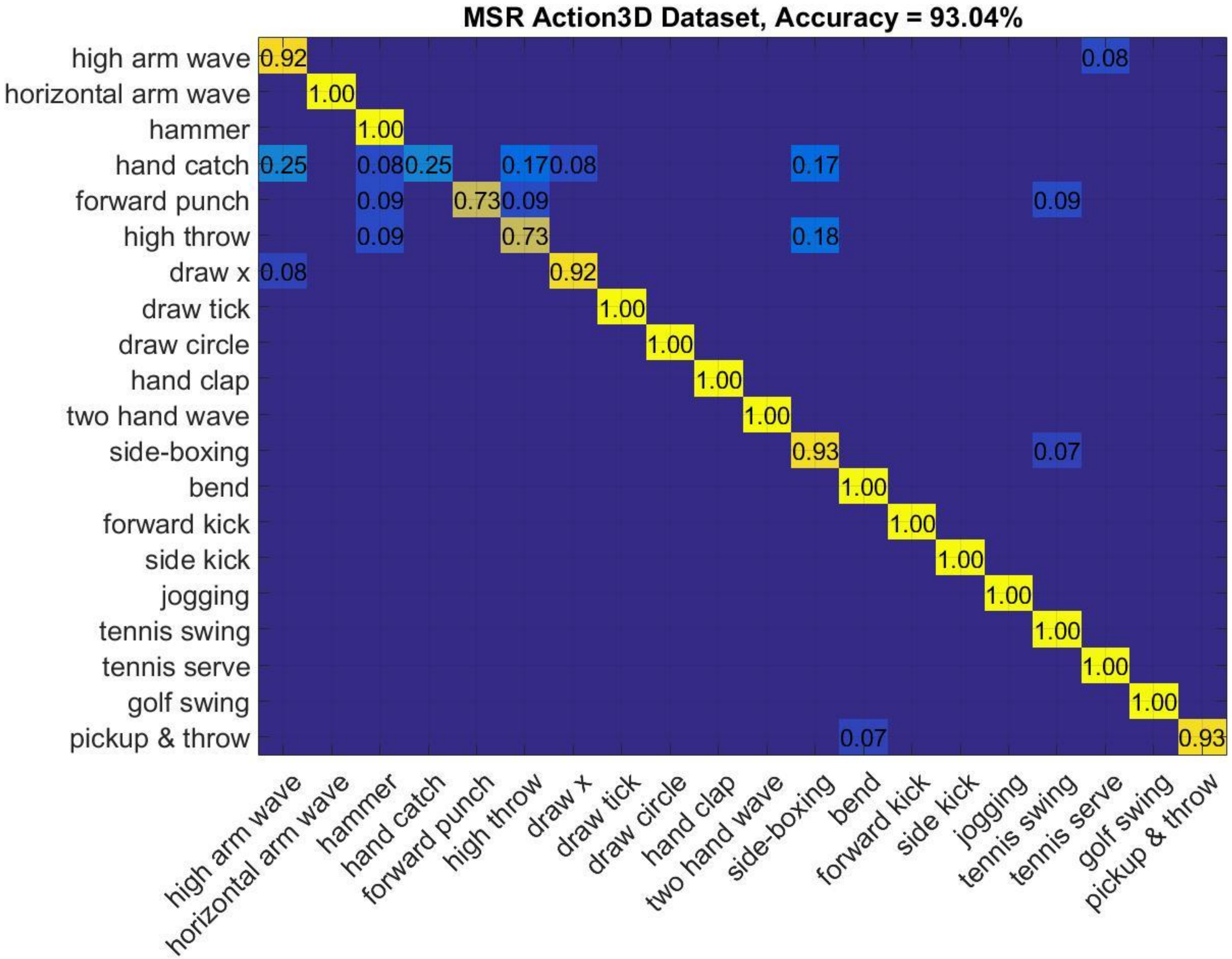


Figure 3. Examples of geodesics between a pair of 3-D curves shown to the left. Two different views of the geodesic are shown to the right.

Experiments --- MSR Action3D Dataset

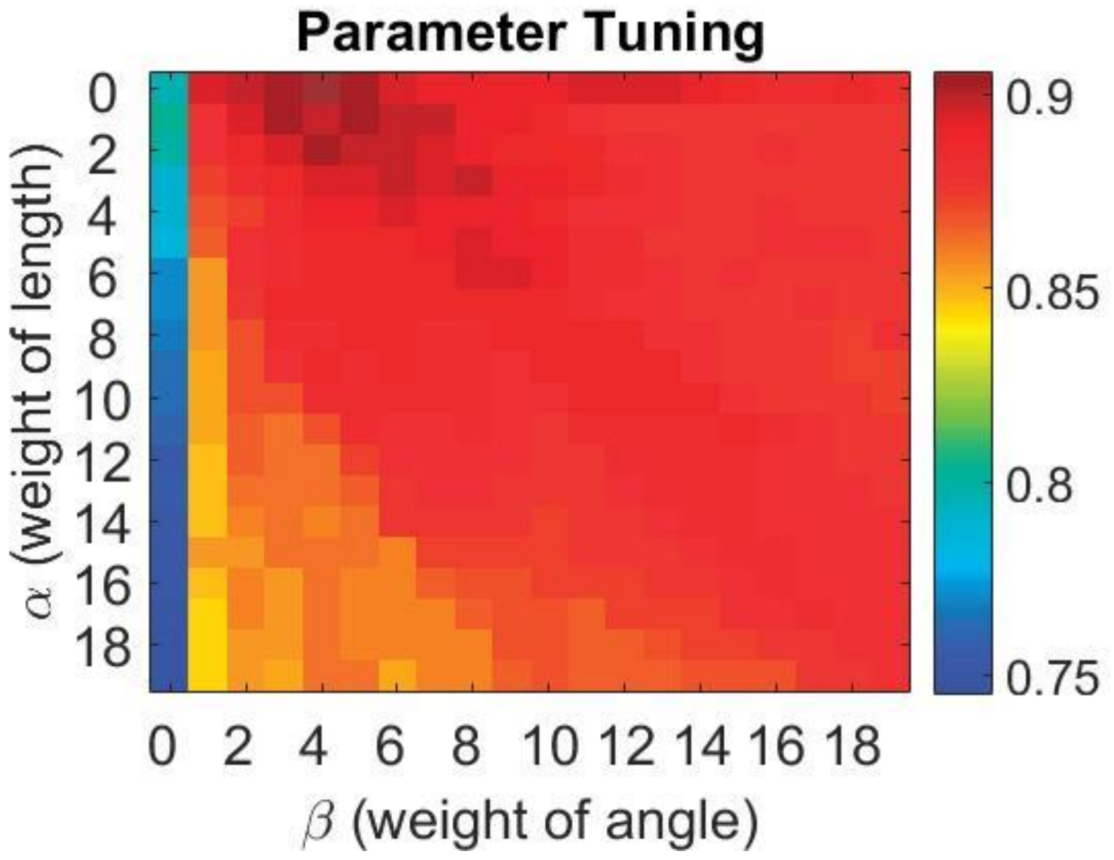


20 actions
10 subjects
2 or 3 instances

Train: subject 1, 3, 5, 7, 9
284 sequences
time 2.45 sec
Test: subjects 2, 4, 6, 8, 10
273 sequences
time 19.45 sec

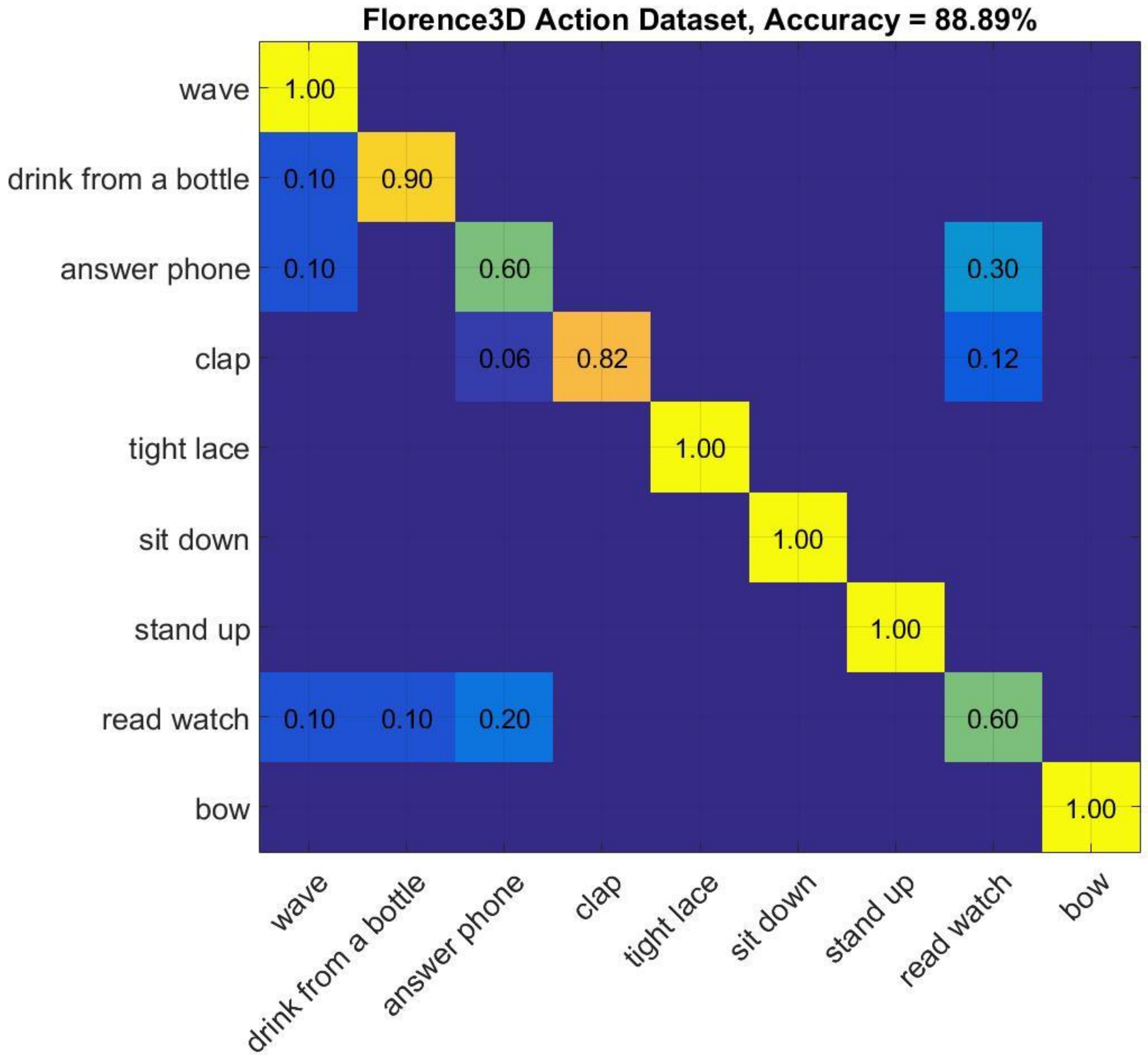
Accuracy: 93.04%

Experiments --- MSR Action3D Dataset



Algorithm	Accuracy (%)	Year
HOPC	91.64	ECCV2014
Lie Algebra	92.46	CVPR2014
Body part + SRVF	92.01	Cyber2015
HON4D	88.89	CVPR2015
Depth Motion Maps-based Local Binary Patterns	91.94	WACV2015
Ours	93.04	---

Experiments --- Florence3D Action Dataset

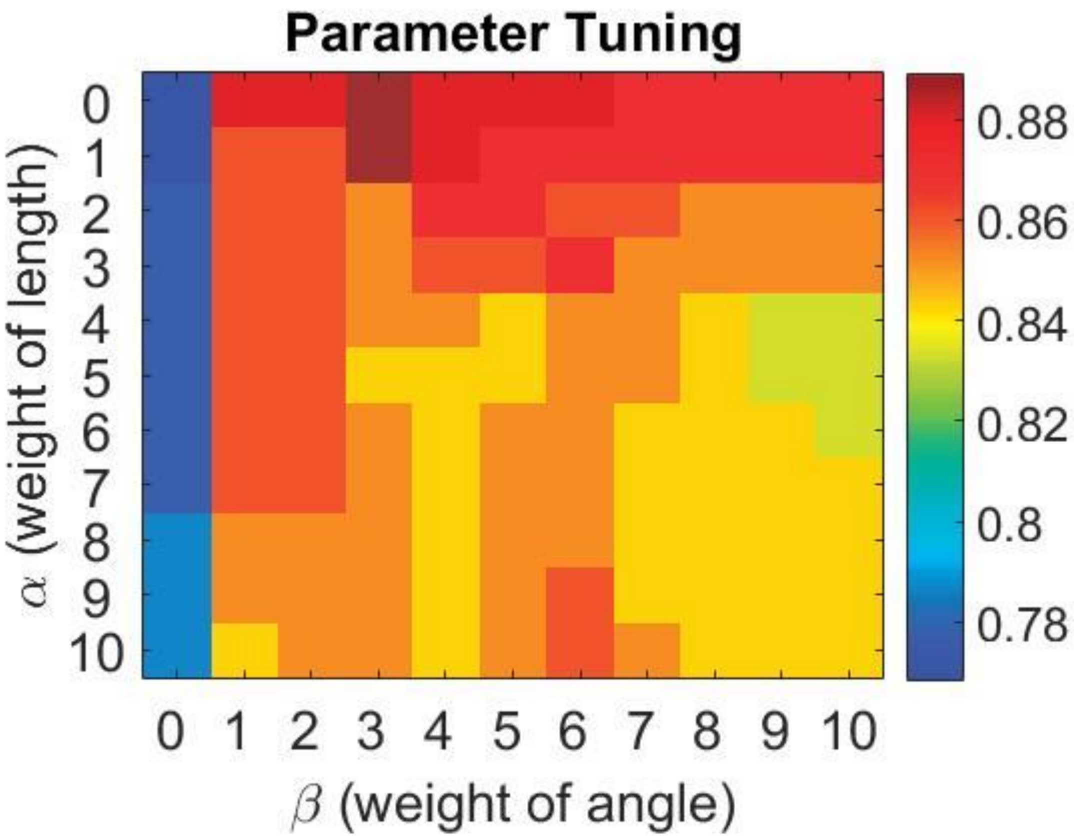


9 actions
10 subjects
2 or 3 instances

Train: subject 1, 2, 3, 4, 5
107 sequences
time 0.72 sec
Test: subjects 6, 7, 8, 9, 10
108 sequences
time 1.54 sec

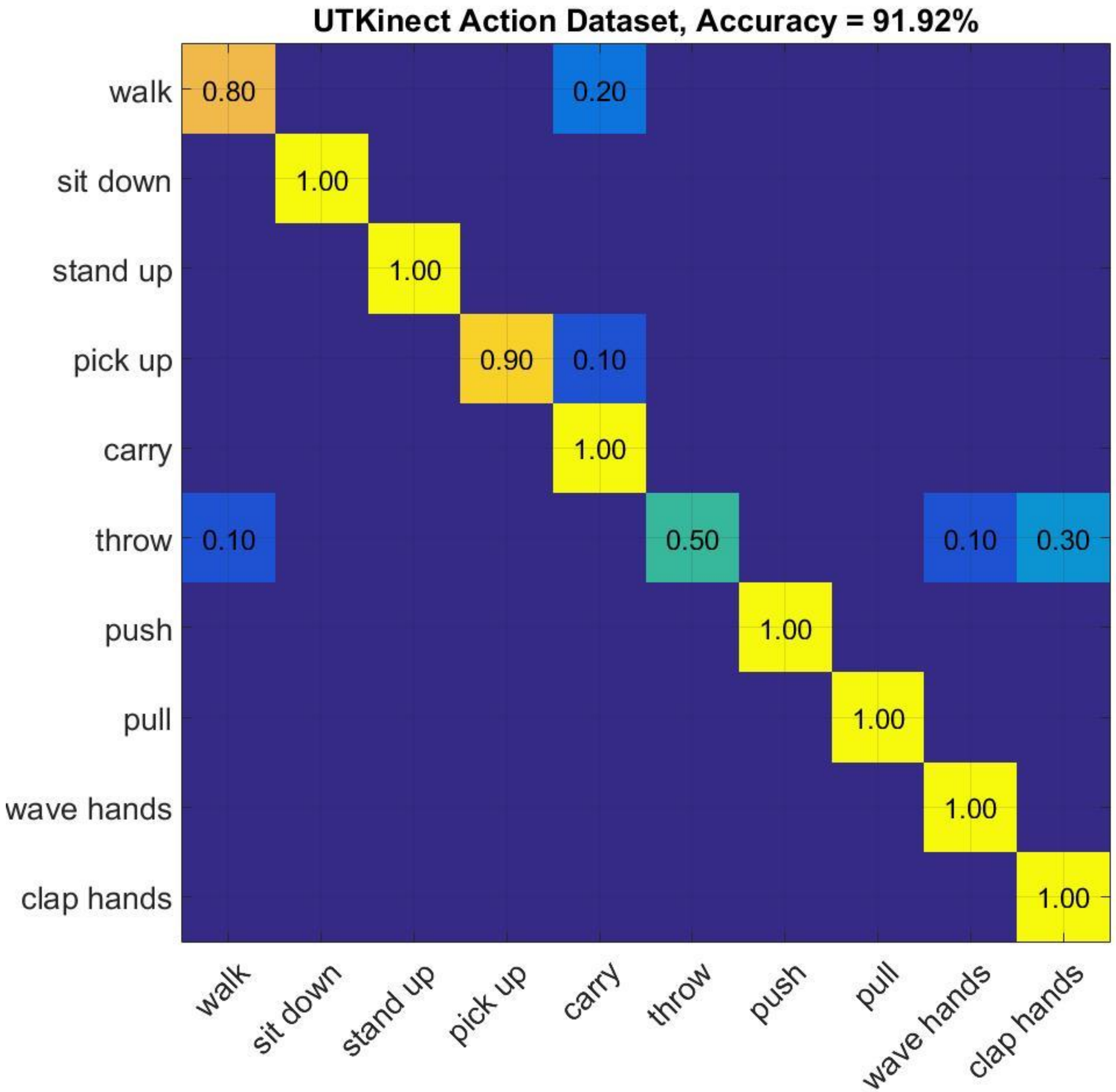
Accuracy: 88.89%

Experiments --- Florence3D Action Dataset



Algorithm	Accuracy (%)	Year
Lie Algebra	86.74	CVPR2014
Body part + SRVF	87.04	Cyber2015
Lie Algebra + SRVF	89.50	CVPR2015
Ours	88.89	---

Experiments --- UTKinect Action Dataset

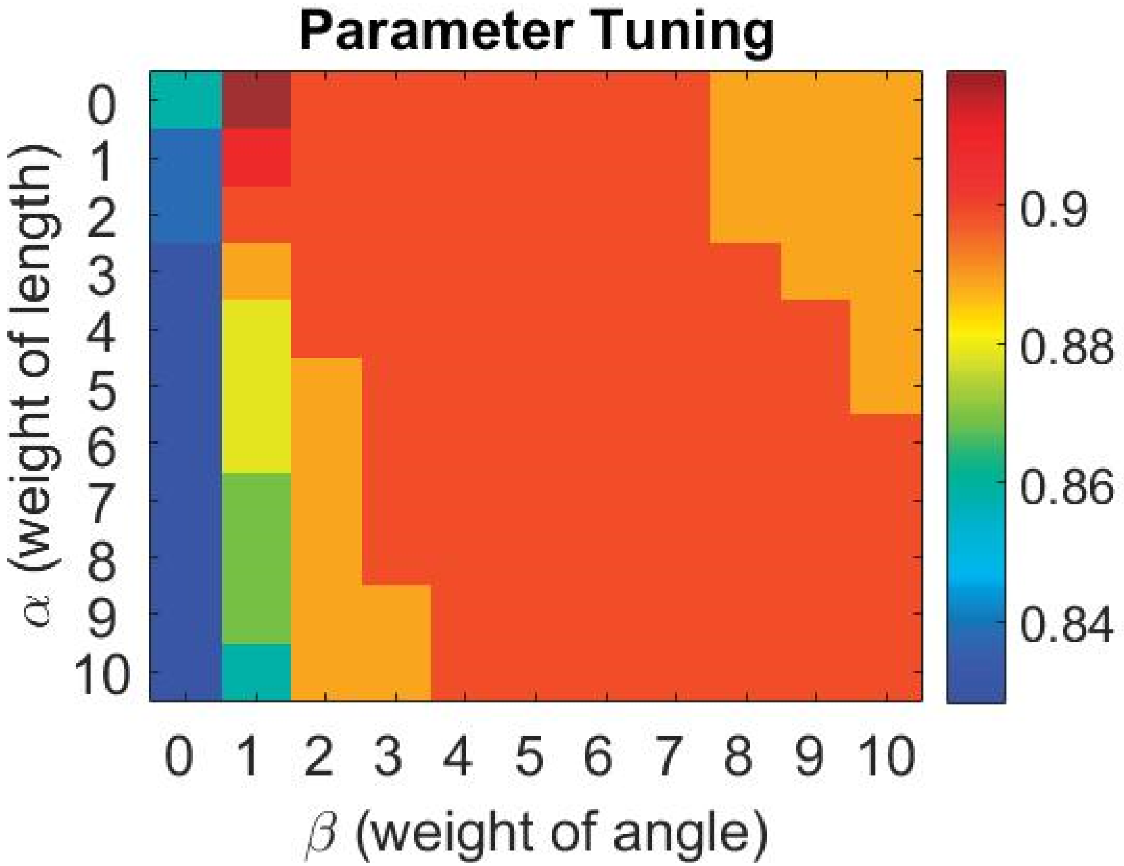


10 actions
10 subjects
2 instances

Train: subject 1, 2, 3, 4, 5
100 sequences
time 1.66 sec
Test: subjects 6, 7, 8, 9, 10
100 sequences
time 3.09 sec

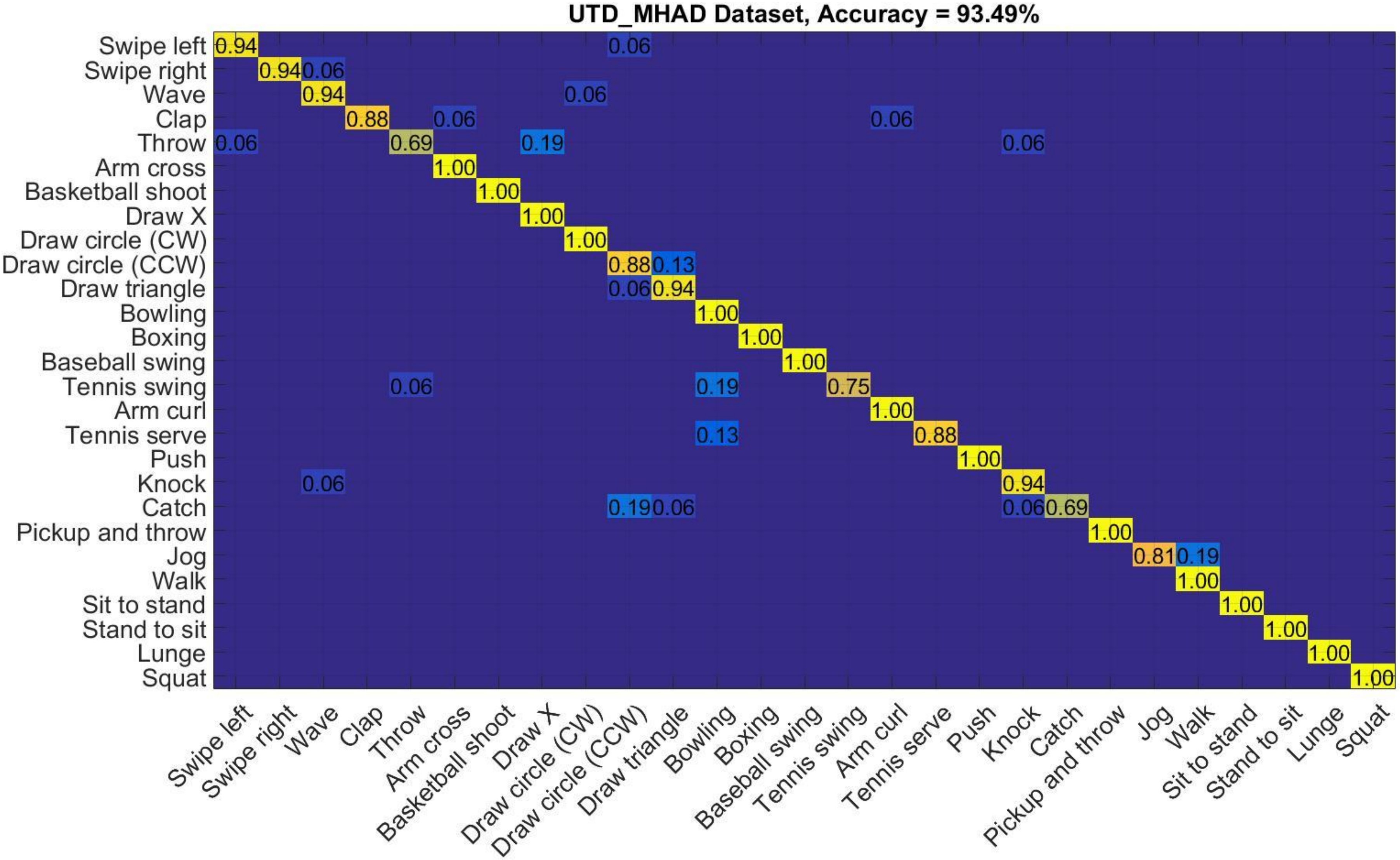
Accuracy: 91.92%

Experiments --- UTKinect Action Dataset



Algorithm	Accuracy (%)	Year
Lie Algebra	92.17	CVPR2014
Body part + SRVF	91.50	Cyber2015
Lie Algebra + SRVF	85.10	CVPR2015
Ours	91.92	---

Experiments --- UTD-MHAD Dataset



27 actions
8 subjects
4 instances

Train: subject 1, 3, 5, 7
431 sequences
time 12.99 sec
Test: subjects 2, 4, 6, 8
430 sequences
time 92.09 sec

Accuracy: 93.49%

Thank You