# Quaternion Space Sparse Decomposition for Motion Compression and Retrieval

*SCA 2012*

Mingyang Zhu, Huaijiang Sun, Zhigang Deng

南京理工大学 Nanjing University of Science and Technology

UNIVERSITY of HOUSTON

# Motivation

- Human motion has sparse nature in both the spatial domain and the temporal domain.
- How to find an efficient way to directly represent intrinsically sparse human motion data in quaternion space.

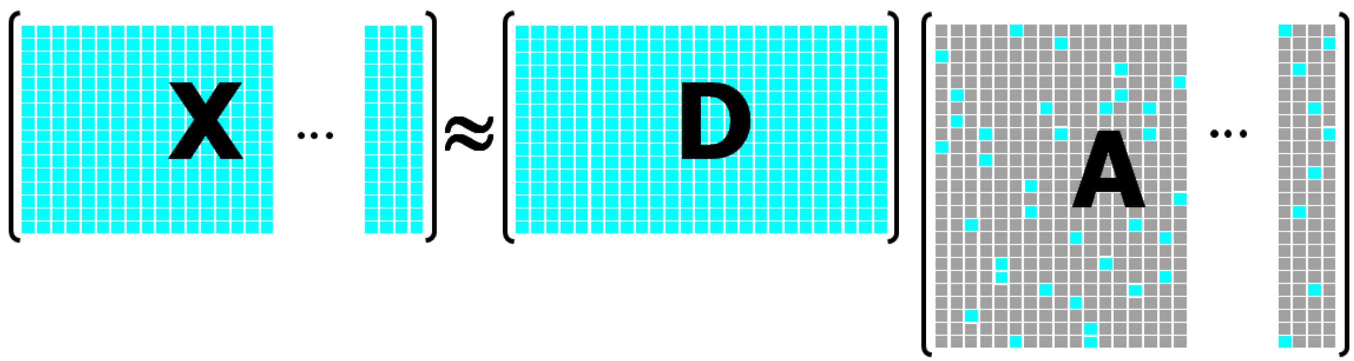
# Related Work-Sparse Representation

- *Sparse coding:*

It calculates corresponding coefficients based on the given signals and dictionary. [MZ93;CBL89;DMZ94;CDS01]

- *Dictionary learning:*

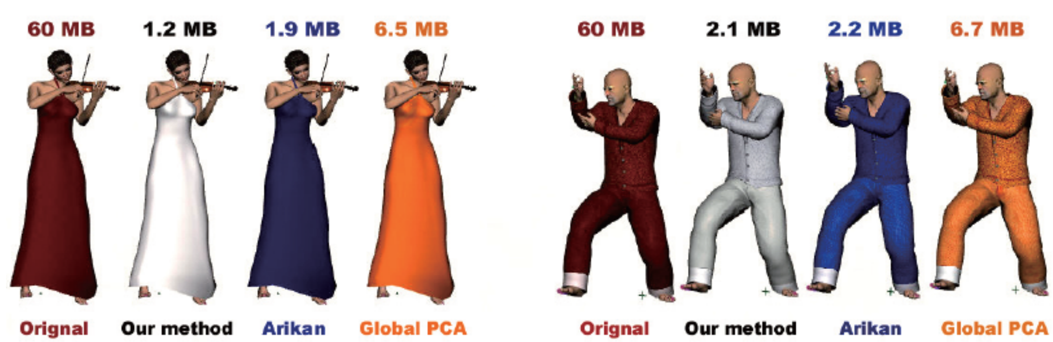
It learns dictionary atoms based on the given signals and the sparse coding coefficients. [ERKD99;AEB06]




Matrix **X** denotes the signal data, Matrix **D** is the dictionary and **A** is the sparse coefficient matrix.

# Related Work-Motion Compression

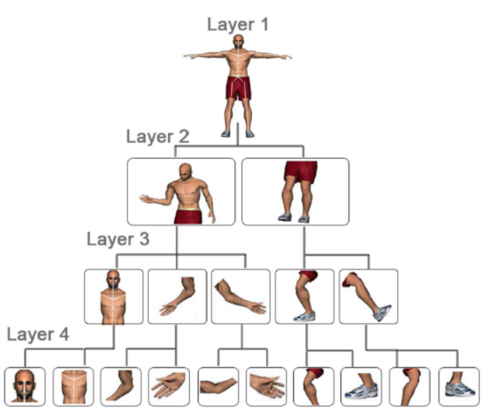
- Human motion compression has been studied by many researchers recently. [LM06;CBL07;BPvdP07]
- Arikan [Ari06] uses spectral clustering and PCA to reduce the data size after motion trajectories are fitted with Bezier curves.
- Gu et al. [GPD09] proposed a pattern indexing scheme for motion capture data compression.
- Tournier et al. [TWC*09] use a principal geodesics analysis (PGA) based inverse kinematics technique to restore the motion.

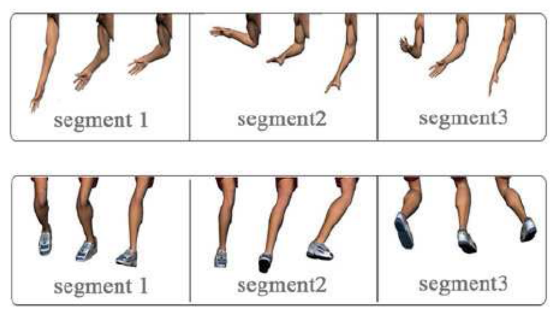



GU, Q., PENG, J., AND DENG, Z. 2008.
**Compression of human motion capture data using motion pattern indexing**. *CGF 2008*

# Related Work-Motion Retrieval

- Human motion retrieval has been a hot topic in recent years with the availability of large-scale motion capture databases. [CMU11;MRC*07]
- Various algorithms were proposed for human motion retrieval: the hierarchical tree for key-frames [LZWP03], the match web structure [KG04], weighted PCA [FF05], a fuzzy search scheme based on the geometric features [MRC05;MR06], the hierarchical motion patterns [DGL09], and low-rank subspace decomposition for motion volume [CS11].




DENG, Z., GU, Q., LI, Q. 2009. **Perceptually Consistent Example-based Human Motion Retrieval.** *I3D 2008*

# Main contributions

- **(1) Quaternion combination.**

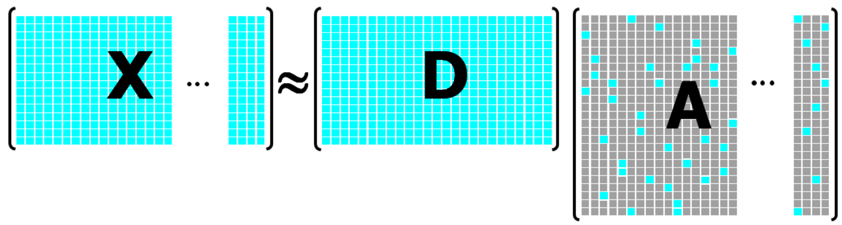
We introduce a novel quaternion space sparse decomposition model by replacing *linear combination* with *quaternion combination*.

- **(2) Motion compression and retrieval application.**

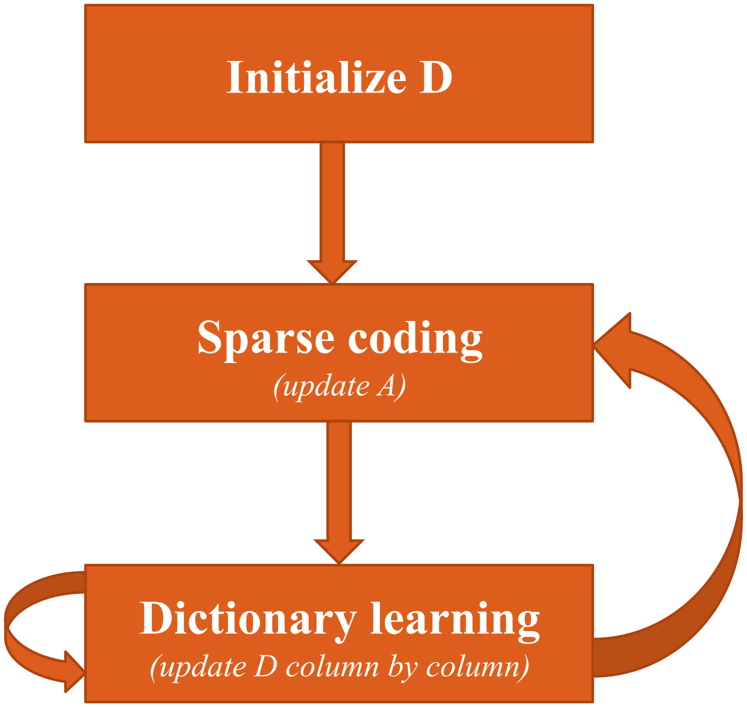
We apply our approach to two selected applications: human motion compression and content-based human motion retrieval.

# Spares representation

- Problem: $\min_{D,A} \|X - DA\|_2^2, \text{s.t.} \forall j \left\|A_j\right\|_0 \leq L$




- Algorithm:
- Step 1: Initialize D
- Step 2: Update A by any match pursuit methods by fixing D
- Step 3: Update D column by column based on SVD method
- Step 4: Repeat step 2,3 until converge



# Quaternion combination

- Linear combination ($DA = D_1A_1 + \cdots + D_nA_n$) does not make sense for quaternions
- We define a meaningful combination between quaternion dictionary D and weight matrix A, such as $D \otimes A = D_1 \odot A_1 \oplus \cdots \oplus D_n \odot A_n$
- In our approach, we define $D_i \odot A_i$ as quaternion power ${\mathrm{D}_i}^{A_i}$ and $D_i \oplus D_j$ as quaternion multiplication.
- However, it cannot be converted to linear form by logarithm rules, because quaternion multiplication is non-commutative.

# Combination order

- However, since quaternion multiplication is non-commutative, the combination order should be given in the quaternion combination.
- In our approach, we define the combination order as the selection order in sparse coding stage.
- Match pursuit methods select dictionary items based on greedy algorithms which means the selection order is the order of sorting weight in a descending manner.

# QSSD Problem

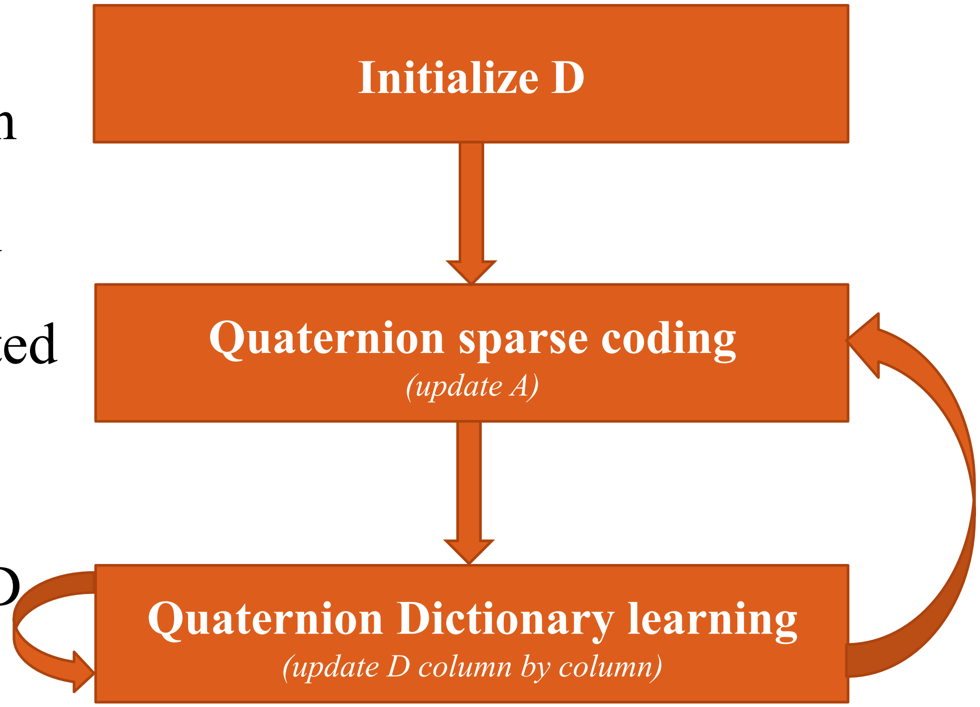
- By using the proposed quaternion combination, the QSSD problem can be stated as follows:

*By given a quaternion signal matrix X, QSSD decomposes X into two parts: quaternion dictionary D and weight matrix A.*

*The quaternion combination ensures the atoms in D are valid quaternions, and weights in A indicates the importance of different atoms for reconstructing given data.*

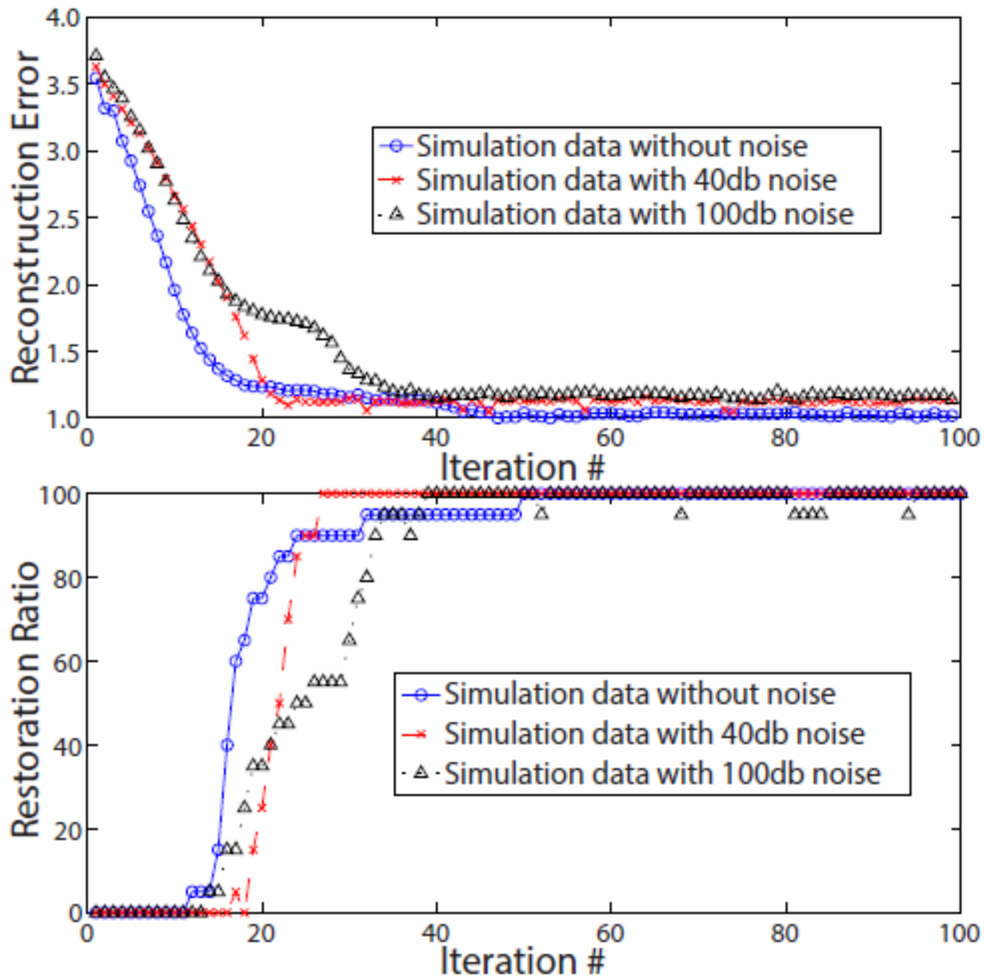
# QSSD algorithm

- Fortunately, there is no quaternion multiplication involved but only quaternion power operation when updating one column of D. That means this stage could be converted into linear problem by logarithm rules.
- Therefore, general K-SVD algorithm is applied for our QSSD problem.



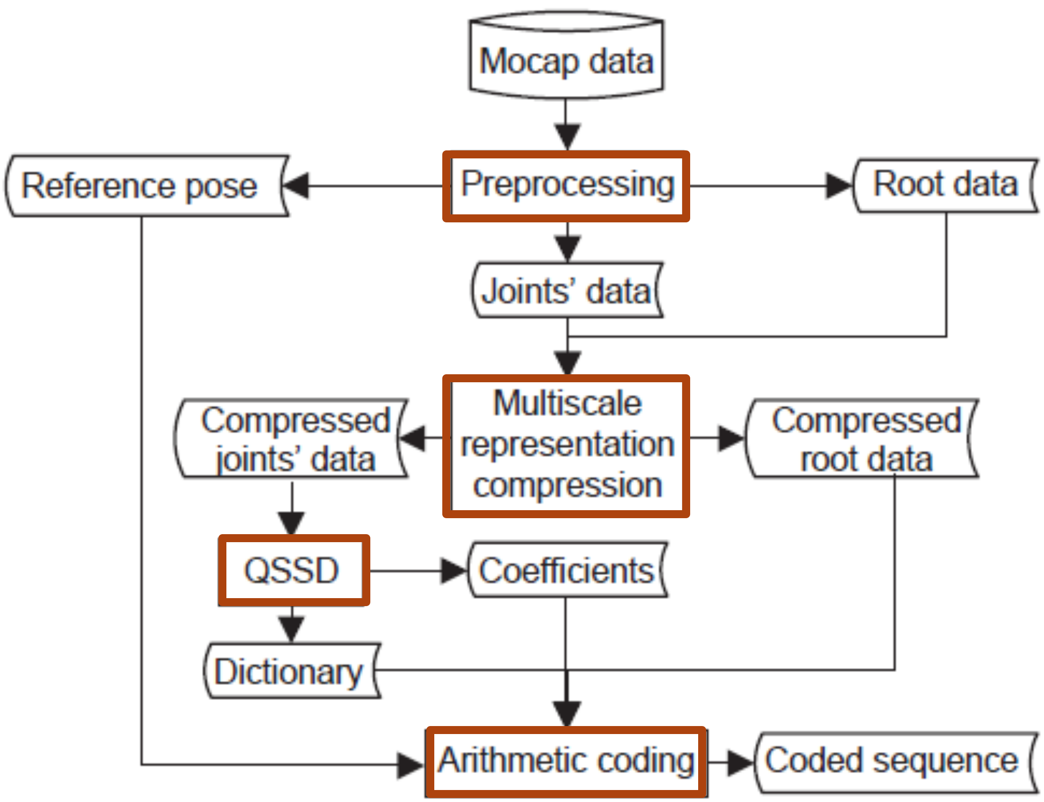

- Step 1: Initialize D (ensure the atoms are valid unit quaternions)
- Step 2: Select atoms by their importance and calculate the corresponding weights
- Step 3: Update dictionary items one by one based on SVD method
- Step 4: Repeat step 2,3 until converge

# Validation via simulation data




- We applied our QSSD algorithm to the generated simulation dataset: the dictionary contained 20 atoms and 500 samples were generated by combining 5 different atoms.

# Application-Motion compression




- Step 1: Preprocessing
- Step 2: Multi-scale representation compression
- Step 3: QSSD decomposition
- Step 4: Arithmetic coding

# Recovered quality

- In our experiment, in order to quantitatively evaluate the quality of recovered (decompressed) motion data, we use two error metrics that employed in previous work: *ARMS* metric [GPD09] and *distortion rate* [Kar04].

$$ARMS = \frac{\sum_{j=1}^{Mkr\#} \sqrt{\left(\sum_{i=1}^{Mkr\#} \left\|P_i - \hat{P}_i\right\|^2\right) / Mkr\#}}{Frame\#}$$

$$DistortionRate = 100 \frac{\left\|P - \hat{P}\right\|}{\left\|P - E(P)\right\|}$$

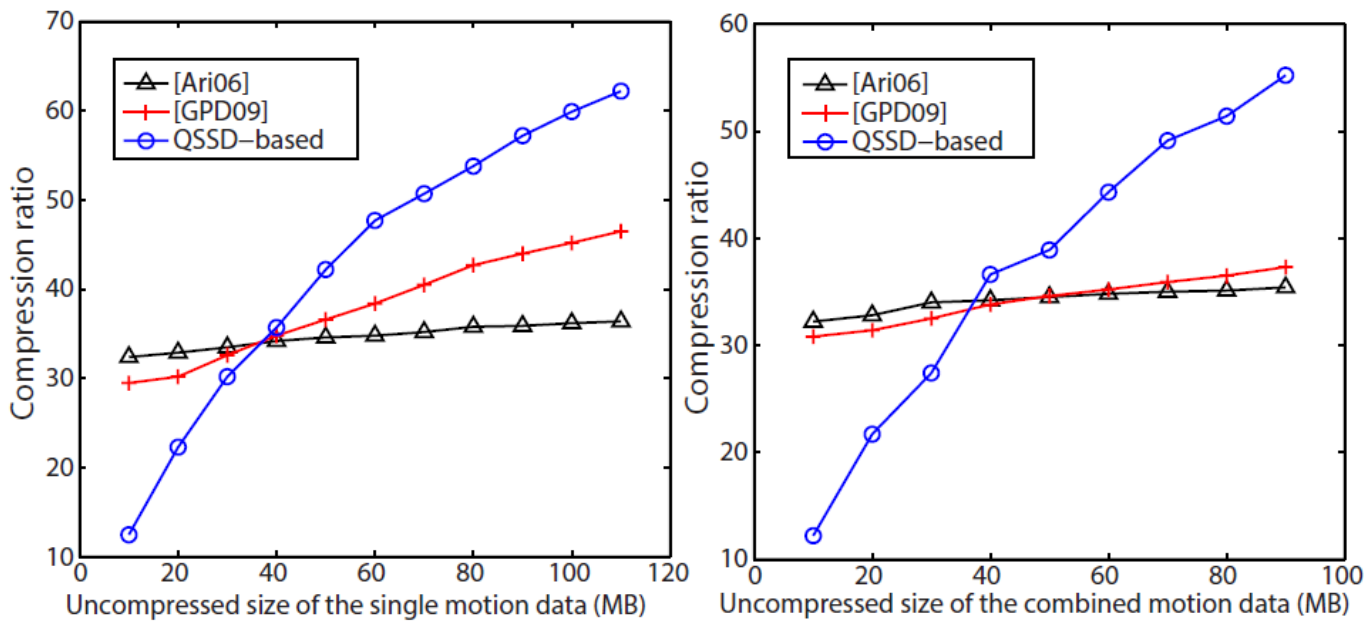
# Comparison results

- We extracted two test datasets from the CMU motion capture database: walk motion dataset and mixed motion dataset.

| Dataset | Parameter | QSSD-based | K-SVD based | [GPD09] | [Ari06] |
|---|---|---|---|---|---|
| Walk motions (Original size: 111MB Total frame #: 145828 Total clip #: 121) | Compressed size(MB) | 1.8 | 1.8 | 2.4 | 3.1 |
| | Compression ratio | 62 | 62 | 46 | 36 |
| | $ARMS$ $(cm)$ | 1.33 | 5.86 | 4.12 | 4.31 |
| | $Distortion$ rate | 2.64 | 6.22 | 5.43 | 5.87 |
| | Compression time $(ms/frm)$ | 42.5 | 23.9 | 2.7 | 0.9 |
| | Decompression time $(ms/frm)$ | 0.8 | 0.7 | 0.7 | 1.0 |
| Mixed motions (Original Size: 93.5MB Total frame #: 122846 Total clip #: 75) | Compressed size(MB) | 1.7 | 1.7 | 2.5 | 2.7 |
| | Compression ratio | 55 | 55 | 37 | 35 |
| | $ARMS$ $(cm)$ | 1.92 | 6.25 | 5.32 | 5.87 |
| | $Distortion$ rate | 2.91 | 8.70 | 6.36 | 6.55 |
| | Compression time $(ms/frm)$ | 48.4 | 25 | 3.1 | 1.0 |
| | Decompression time $(ms/frm)$ | 0.7 | 0.6 | 0.7 | 1.2 |

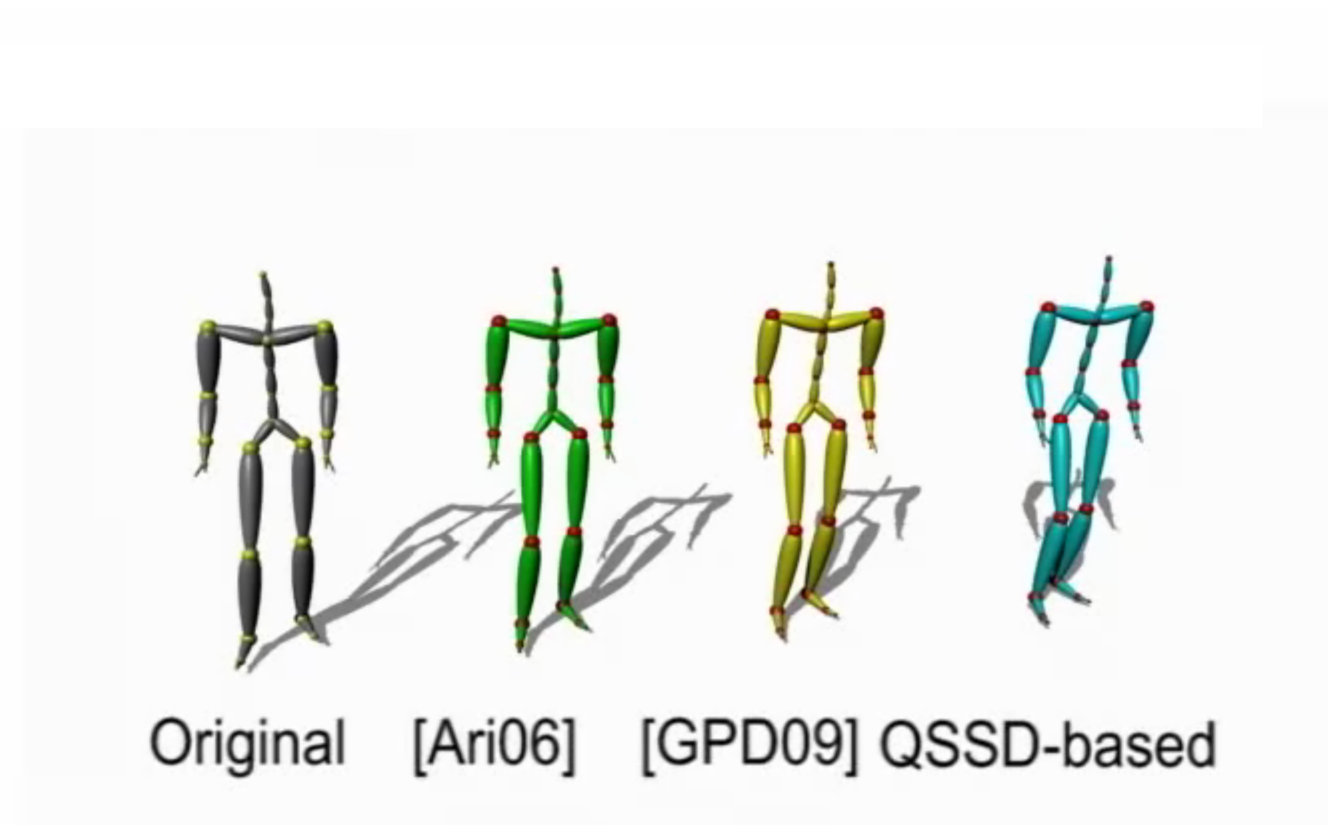
# Compression ratio curves

- We compared our approach with the other two comparative approaches in terms of compression ratio with different size of motions.



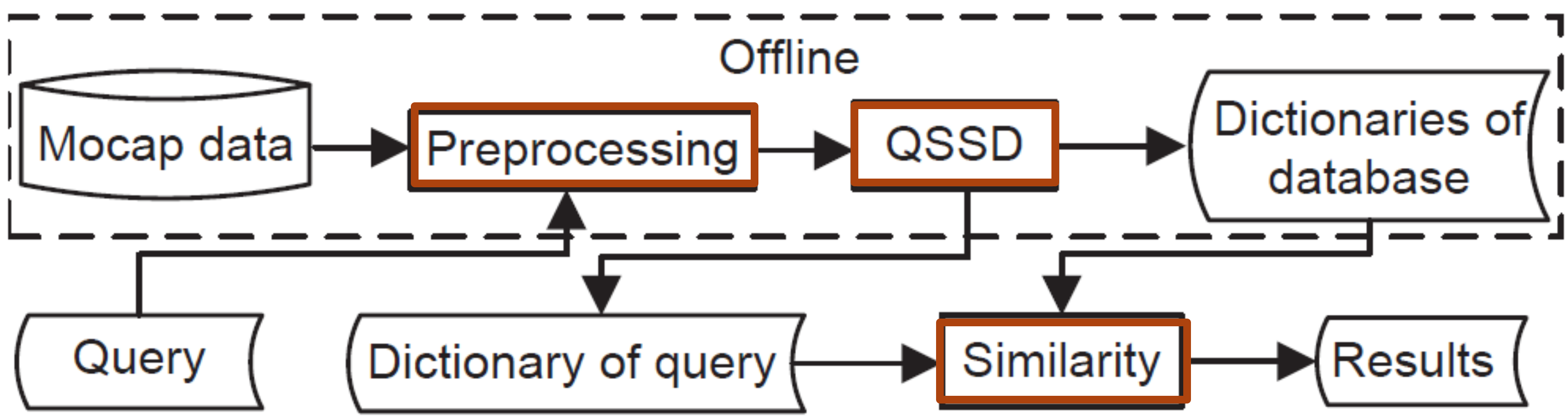

***Left:*** walking motion dataset; ***Right*:** mixed motion dataset

# Review of recovered motions



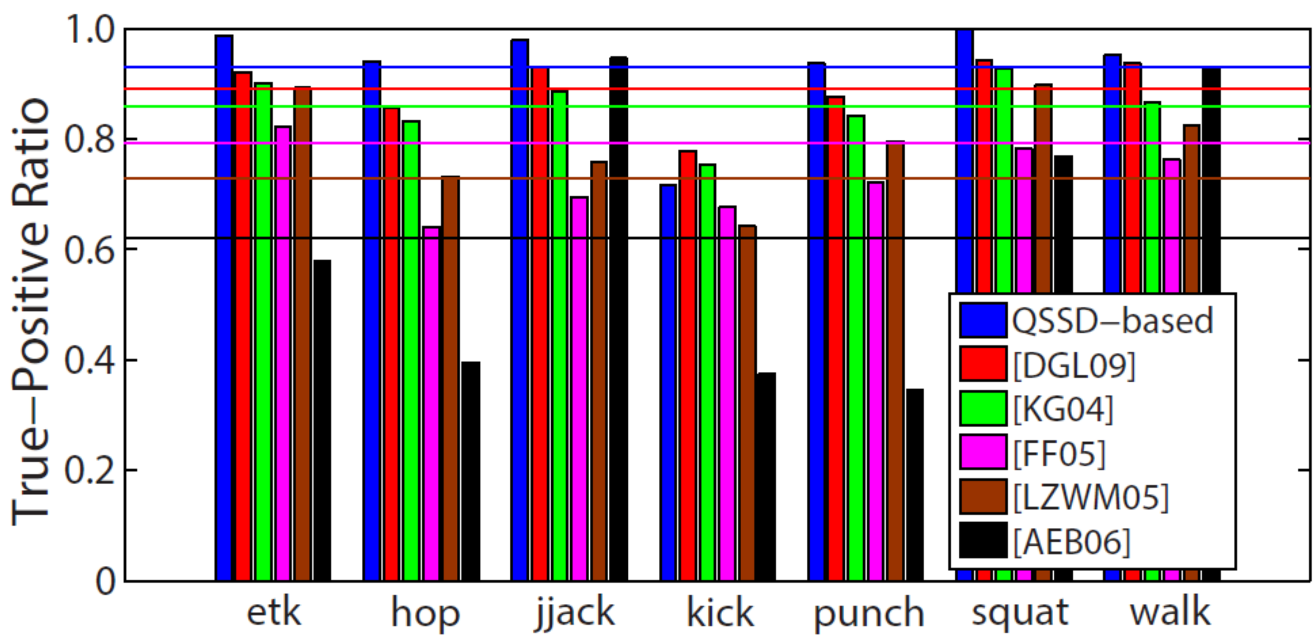
# Application-Motion retrieval

- **Working pipeline**
- Step 1: Preprocessing
- Step 2: QSSD for each motion data
- Step 3: Similarity computing based on dictionaries
- Step 4: Result ranking



# Search accuracy

- We conducted an accuracy evaluation experiment by using each motion in the dataset as the query, then compute a true-positive ratio which is defined as the true percentage of the *topN(=20)* results. Finally, the average true-positive ratio of 7 motion categories are calculated.




Six horizontal lines illustrate the averaged true-positive ratios by the six approaches.

# Confusion matrix

- Confusion matrix is a widely used criterion to evaluate a classification algorithm. We conducted the experiment to reveal the confusion level between any two categories of motions of our approach.

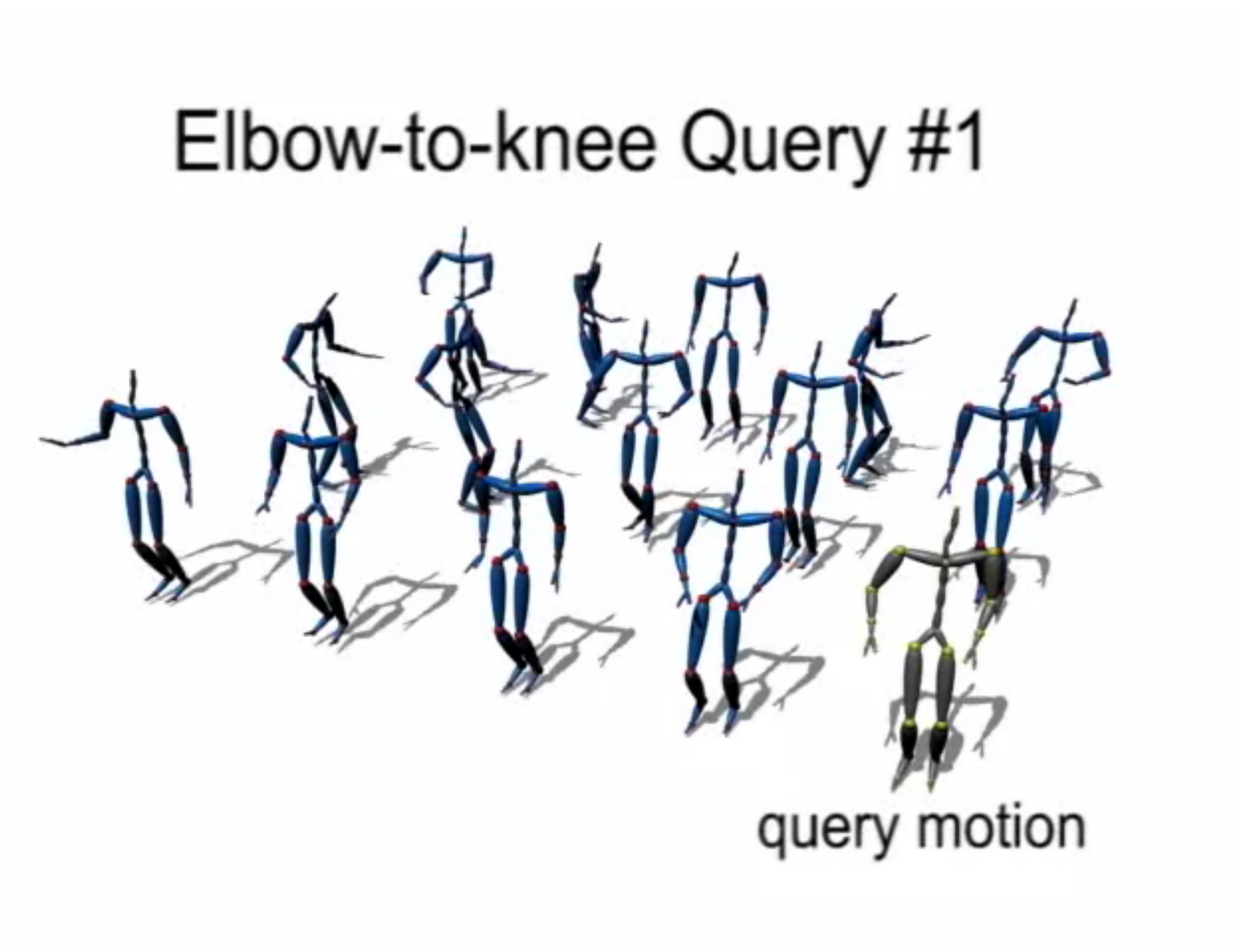
| | etk | hop | jjack | kick | punch | squat | walk |
|---|---|---|---|---|---|---|---|
| etk | 0.69 | 0.09 | 0.02 | 0.02 | 0.02 | 0.01 | 0.14 |
| hop | 0.06 | 0.45 | 0.06 | 0.05 | 0.09 | 0.02 | 0.27 |
| jjack | 0.01 | 0.02 | 0.95 | 0.00 | 0.01 | 0.01 | 0.01 |
| kick | 0.09 | 0.30 | 0.00 | 0.43 | 0.06 | 0.00 | 0.13 |
| punch | 0.07 | 0.15 | 0.04 | 0.11 | 0.35 | 0.15 | 0.11 |
| squat | 0.01 | 0.03 | 0.08 | 0.00 | 0.06 | 0.78 | 0.04 |
| walk | 0.01 | 0.02 | 0.00 | 0.00 | 0.02 | 0.00 | 0.95 |

| | etk | hop | jjack | kick | punch | squat | walk |
|---|---|---|---|---|---|---|---|
| etk | 0.99 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.01 |
| hop | 0.01 | 0.94 | 0.04 | 0.00 | 0.00 | 0.00 | 0.01 |
| jjack | 0.00 | 0.00 | 0.98 | 0.00 | 0.02 | 0.00 | 0.00 |
| kick | 0.01 | 0.00 | 0.01 | 0.72 | 0.01 | 0.03 | 0.23 |
| punch | 0.00 | 0.00 | 0.05 | 0.00 | 0.94 | 0.00 | 0.01 |
| squat | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 1.00 | 0.00 |
| walk | 0.01 | 0.00 | 0.01 | 0.03 | 0.00 | 0.00 | 0.95 |

***Left:*** *the confusion matrix of K-SVD on different motions*

***Right:*** *the confusion matrix of our approach on different motions*

# Review of motion retrieval results



# Discussion and Conclusions (1)

- We introduce a novel quaternion space sparse decomposition (QSSD) model that decomposes rotational human motion data into a dictionary part and a weight part.

Features:

(1) More robust for decomposing rotational data by using quaternion combination to replace linear combination.

(2) Benefit for human motion compression and retrieval.

# Discussion and Conclusions (2)

- **Limitations:**

(1) The QSSD model takes significant computational time, since the expensive computational cost of quaternion combination.

(2) The decomposition results are less intuitive for certain applications such as human motion editing/synthesis, because some important constraints (e.g., non-negativity and affinity) were not applied.

# Thank you!

Nanjing University of Science and Technology

UNIVERSITY of HOUSTON