Outline

Introduction

Framework

Algorithm

Experiment

Conclusion

Key Frames Hysteresis-Seeking Based on Motion Change Points for RGB-D Video

Hui Feng

collaborated with Yong Nie, Peng Zhang and Bo Hu

Digital Signal Processing and Transmission Lab

Fudan University

Dec. 16, 2015

Outline

- Introduction
- Key-Frame Extraction Framework
  - Energy Threshold
  - Motion Change Point Detection
  - Frame Indices Fusion
- Extrema Hysteresis-Seeking Algorithm
- Experimental Result
- Conclusion

Outline
Introduction
Framework
Algorithm
Experiment
Conclusion

Introduction – Key frame

Outline

Introduction

Framework

Algorithm

Experiment

Conclusion

Key-frame (or key pose or key actionlet) extraction becomes a focused problem of human action recognition in recent years.

- Effectiveness[1]
- Efficiency[2]
- Action understanding[3]

Figure 1 : Examples of actions from KTH. Note that even a single frame is often sufficient to recognise what a person is doing.(reprint from [1])

---

[1] K. Schindler and L. Van Gool. "Action snippets: How many frames does human action recognition require?" In: *2008 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2008, pp. 1–8. DOI: 10.1109/CVPR.2008.4587730.

[2] N. Azouji and Z. Azimifar. "A new approach to speed up in action recognition based on key-frame extraction". In: *2013 8th Iranian Conference on Machine Vision and Image Processing (MVIP)*. 2013, pp. 219–222. DOI: 10.1109/IranianMVIP.2013.6779982.

[3] Yongxiong Wang and Yubo Shi. "Human activities segmentation and location of key frames based on 3D skeleton". In: *2014 33rd Chinese Control Conference (CCC)*. 2014, pp. 4786–4790. DOI: 10.1109/ChiCC.2014.6895749.

Introduction – RGB-D video

Outline

Introduction

Framework

Algorithm

Experiment

Conclusion

Why RGB-D videos?

- Affordable RGB-D sensors
  - Microsoft Kinect, PrimeSense PSDK, ASUS Xtion Pro and Pro Live
- High-level feature: skeleton
  - The effectiveness of joint feature for action recognition has been studied in [4]

(a) Microsoft kinect **(b)** ASUS Xtion

Figure 2 : RGB-D sensors

[4] Hueihan Jhuang et al. "Towards Understanding Action Recognition". In: *2013 IEEE International Conference on Computer Vision (ICCV)*. 2013, pp. 3192–3199. DOI: 10.1109/ICCV.2013.396.

Introduction – Existing method

Outline

Introduction

Framework

Algorithm

Experiment

Conclusion

- [5] considers key frames
  - the representative frames that include a set of salient images
  - the combination of low-level features as the criterion.
- [6] locates the key frames
  - the sphere of maximum energy information
  - segment the action video temporally and group the atomic action units iteratively

---

[5] N. Azouji and Z. Azimifar. "A new approach to speed up in action recognition based on key-frame extraction". In: *2013 8th Iranian Conference on Machine Vision and Image Processing (MVIP)*. 2013, pp. 219–222. DOI: 10.1109/IranianMVIP.2013.6779982.

[6] Yongxiong Wang and Yubo Shi. "Human activities segmentation and location of key frames based on 3D skeleton". In: *2014 33rd Chinese Control Conference (CCC)*. 2014, pp. 4786–4790. DOI: 10.1109/ChiCC.2014.6895749.

Introduction – Existing method

Outline

Introduction

Framework

Algorithm

Experiment

Conclusion

- Our method:
  - utilizes the high-level joint data feature
  - locates key frames directly by motion change points seeking
- Contributions:
  - A new key-frame extraction framework for RGB-D videos based on motion change points.
  - A hysteresis extrema seeking algorithm to capture motion change points robustly.

Key-Frame Extraction Framework

Outline

Introduction

Framework

Algorithm

Experiment

Conclusion

Framework includes:

- Energy Threshold
- Motion Change Point Detection
- Frame Indices Fusion

Data representation

- By [7], the body skeleton data of RGB-D action video can be obtained as[8]:

$$V_i = [f_1^i, f_2^i, \cdots, f_{n_i}^i]^T, \tag{1}$$

$$f_t^i = [s_1^{i,t}, s_2^{i,t}, \cdots, s_{N_{joint}}^{i,t}]^T, \tag{2}$$

$$s_j^{i,t} = [x^{i,t,j}, y^{i,t,j}, z^{i,t,j}]^T \tag{3}$$

[7] Sean Kean, Jonathan C. Hall, and Phoenix Perry. "Microsofts Kinect SDK". . In: *Meet the Kinect* (2011), pp. 151–173.

[8] Without confusion, the superscript or subscript may be omitted in the following.

Key-Frame Extraction Framework

Outline

Introduction

Framework

Algorithm

Experiment

Conclusion

- Skeleton model

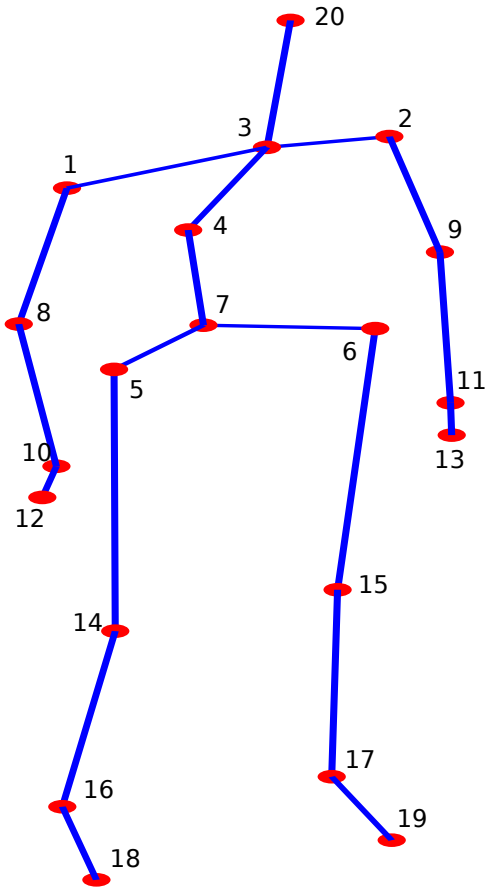

Figure 3 : Skeleton model for 3D action videos

Energy Threshold

Outline
Introduction
Framework
Algorithm
Experiment
Conclusion

Why need energy threshold?

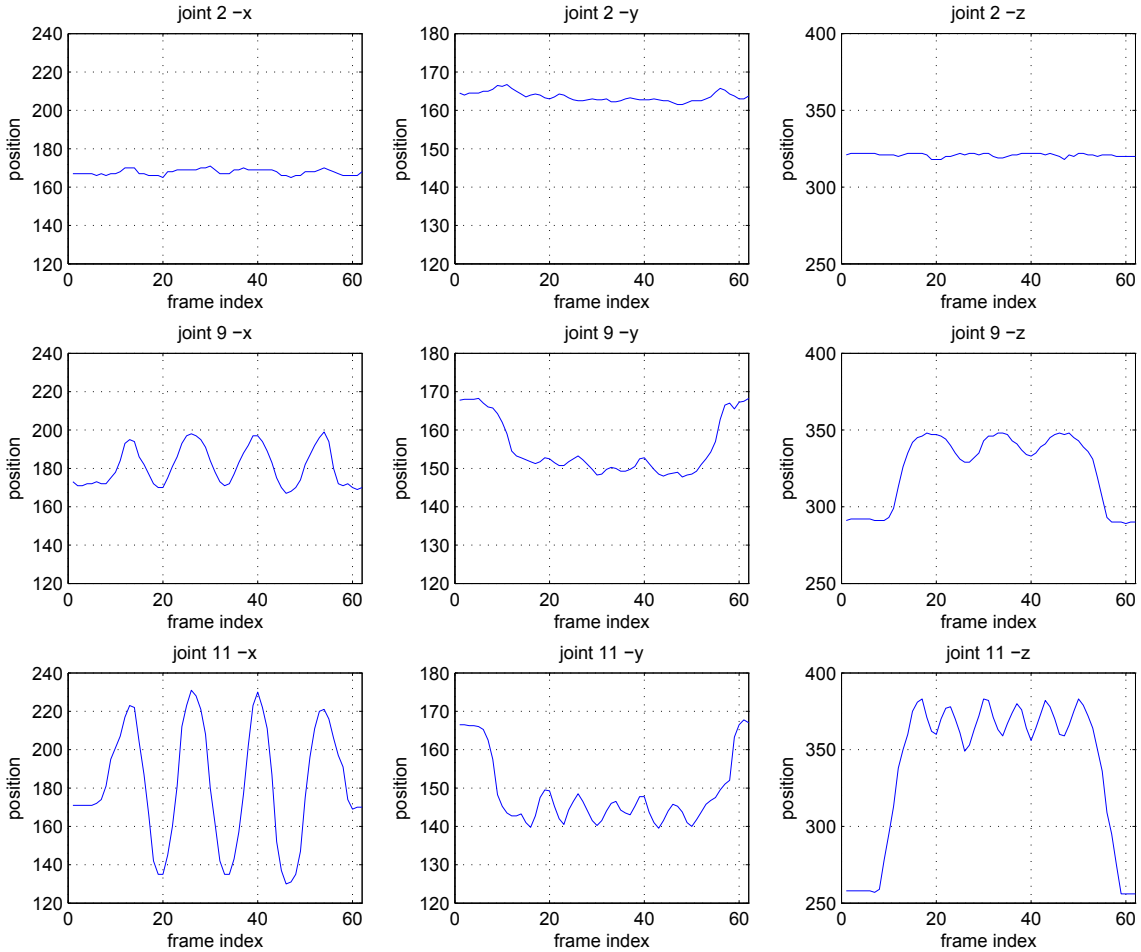

Figure 4 : Action *waving* example: motion position sequence data for joint-$2, 9, 11$ (top to bottom) along axis-x, y, z (left to right) respectively. Obviously, movement of the joint-2 is ignorable compared to joint-11, and the movement of joint-11 along axis-y is ignorable compared to axis-x.

For more details about the action video or how joints arranged, please refer to *MSR Action3D Dataset*

Energy Threshold

Outline
Introduction
Framework
Algorithm
Experiment
Conclusion

- Energy function
  The energy for the movement of one joint along one axis:

$$E_d^s = \sum_{t=1}^{n_i-1} (s^{t+1}(d) - s^t(d))^2 \tag{4}$$

  where $d = 1, 2, 3$ denotes x, y, z axis respectively, $s = s_1, s_2, \cdots, s_{N_{joint}}$.
- Energy thresh
  For video i:

$$E_{max}^i = \max_{s,d} E_d^s, \tag{5}$$

$$E_{thresh}^i = p \cdot E_{max}^i \tag{6}$$

  where p is the proportional constant.
  The movements whose energy is less than E_{thresh}^i will be *filtered out*.

Motion Change Point Detection

Outline

Introduction

Framework

Algorithm

Experiment

Conclusion

- Motion change points

  The positions where the waveform tendency changes.

- Problem: there might be jitters with the waveform.

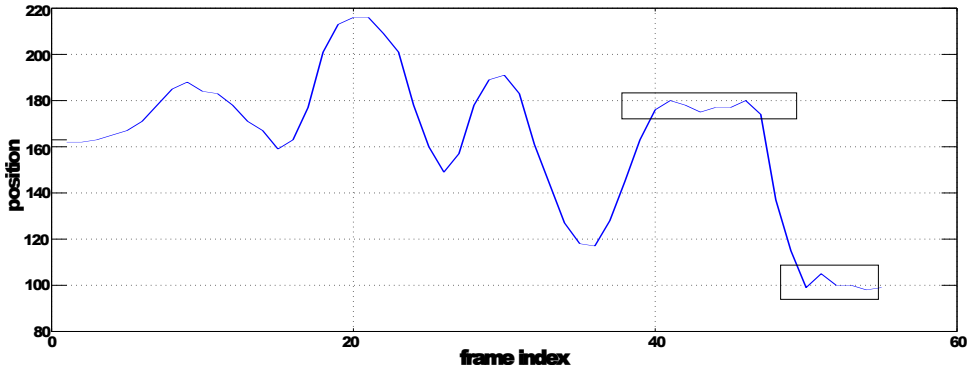

Figure 5 : waveform with jitters

- Algorithm to detect change points robustly

  we develop the hysteresis extrema seeking algorithm,

  which will be explained in detail in subsection *Algorithm*.

Frame Indices Fusion

Outline

Introduction

Framework

Algorithm

Experiment

Conclusion

Table 1 shows an example of key frames indices list found by the steps before.

Table 1 : Key frame indices list

frame index	$\cdots$	43	44	45	46	47	48	49	50	51	52	53	$\cdots$
indicator	$\cdots$	0	0	1	0	0	0	0	0	1	1	0	$\cdots$

- Adjacent key frames around frame index 51.
- DBSCAN[9] to cluster the adjacent motion change point indices in the same body part[10].
- Finally, the motion change point frame indices in different body parts are combined to obtain the final indices for key frames.

[9] Martin Ester et al. "A density-based algorithm for discovering clusters in large spatial databases with noise". In: *Proceedings of International Conference on Knowledge Discovery & Data Mining* (1996), pp. 226–231.

[10] In this paper, the human body skeleton is divided into body torso and four limb parts.

Hysteresis Extrema Seeking

Outline

Introduction

Framework

Algorithm

Experiment

Conclusion

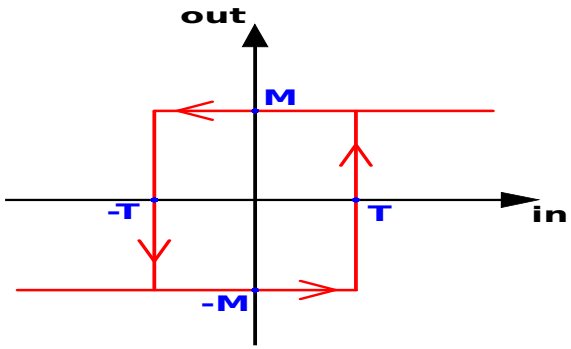

Figure 6 : Typical hysteresis curve

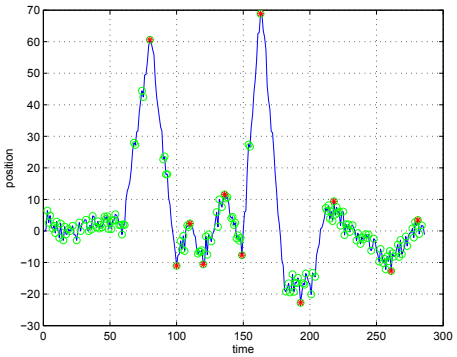

Figure 7 : Waveform with jitters

Hysteresis comparator Inspiration

- Small jitters will not cause the comparator's state to switch.
- The same mechanism in extrema searching.

Definitions I

Assuming waveform $W = [p_1, p_2, p_3, \cdots, p_L]$, let

$$D = derivative(W), \text{where} \tag{7}$$

$$D(i) = \begin{cases} 0 & , \text{for } i = 1 \\ W(i) - W(i-1) & , \text{for } i > 1 \end{cases} \tag{8}$$

$$I = integral(D) = W - W(1) \cdot \mathbf{1} \tag{9}$$

where $\mathbf{1}$ is a $1 \times L$ vector of all 1.

- **Baselines**: the reference values for calculating the accumulated variation of waveform.
  - B_{low} , for calculating positive accumulated variation
  - B_{high} , for calculating negative accumulated variation

  The baselines may be updated along with time t. It always holds: $B_{low} \leq B_{high}$.

Definitions II

- **States and threshold**
  - Define two states for waveform:
    1. positive state: up tendency S_{up};
    2. negative state: down tendency S_{down}.
  - Like the hysteresis comparator, we need to set a threshold THD for tendency state switching.
  - Offset Δ: from the *baseline*(s) to integral value I as the quantities, which represent the accumulated variations of the waveform, to compare with the threshold THD. Specifically, at time t,

$$\Delta_{pos}(t) = I(t) - B_{low} \tag{10}$$

$$\Delta_{neg}(t) = I(t) - B_{high} \tag{11}$$

Definitions III

- **State determining condition**: the tendency of waveform at time t is determined by:

$$S(t) = \begin{cases} S_{up} & \text{, if } \Delta_{pos}(t) >= THD \\ S_{down} & \text{, if } \Delta_{neg}(t) <= -THD \\ S(t-1) & \text{, else} \end{cases} \tag{12}$$

- **Baseline update operation**
  - Initially, both baselines are set 0, $B_{low} = B_{high} = 0$, and the initial tendency of waveform is undetermined.

Outline
Introduction
Framework
Algorithm
Experiment
Conclusion

Definitions IV

- At the stage of determining the initial tendency of waveform, the baselines will be updated by:

$$B_{high} = I(t), \text{ if } S(t) \text{ not determined, } I(t) > B_{high} \quad (13)$$

$$B_{low} = I(t), \text{ if } S(t) \text{ not determined, } I(t) < B_{low} \quad (14)$$

$$B_{high} = B_{low} = I(t), \text{ if } S(t) \text{ determined.} \quad (15)$$

- Once the initial tendency of waveform is determined, there will always exist $S(t-1) \in \{S_{up}, S_{down}\}$, then the baselines will be updated by:

$$B_{high} = I(t), \text{ if } S(t-1) = S_{up}, I(t) > B_{high} \quad (16)$$

$$B_{low} = I(t), \text{ if } S(t-1) = S_{up}, S(t) = S_{down} \quad (17)$$

$$B_{low} = I(t), \text{ if } S(t-1) = S_{down}, I(t) < B_{low} \quad (18)$$

$$B_{high} = I(t), \text{ if } S(t-1) = S_{down}, S(t) = S_{up} \quad (19)$$

Outline
Introduction
Framework
Algorithm
Experiment
Conclusion

Algorithm I

HYSTERESIS EXTREMA SEEKING ALGORITHM

Input: waveform W, threshold THD

Output: set of extrema indices E

Initialization :

$B_{low} = B_{high} = 0, t_{initial} = L, E = \emptyset$

Determine the initial tendency of the waveform

for $t = 1$ to $L - 1$ **do**

  check tendency by (12) and update baselines by (13)-(15), record the index if baselines are updated

  if initial tendency determined **then**

    $t_{initial} = t$; break;

  end if

end for

LOOP Process

Algorithm II

8: **for** $t = t_{initial} + 1$ to $L - 1$ **do**
9: check tendency by (12) and update baselines by (16)-(19), record the index if baselines are updated
10: **if** tendency changed **then**
11: $E = E \cup t_{last\_updated}$ ($t_{last\_updated}$, the last updated index of the referred baseline)
12: **end if**
13: **end for**
14: **return** E

Outline
Introduction
Framework
Algorithm
Experiment
Conclusion

Experiment for extrema seeking

Outline
Introduction
Framework
Algorithm
Experiment
Conclusion

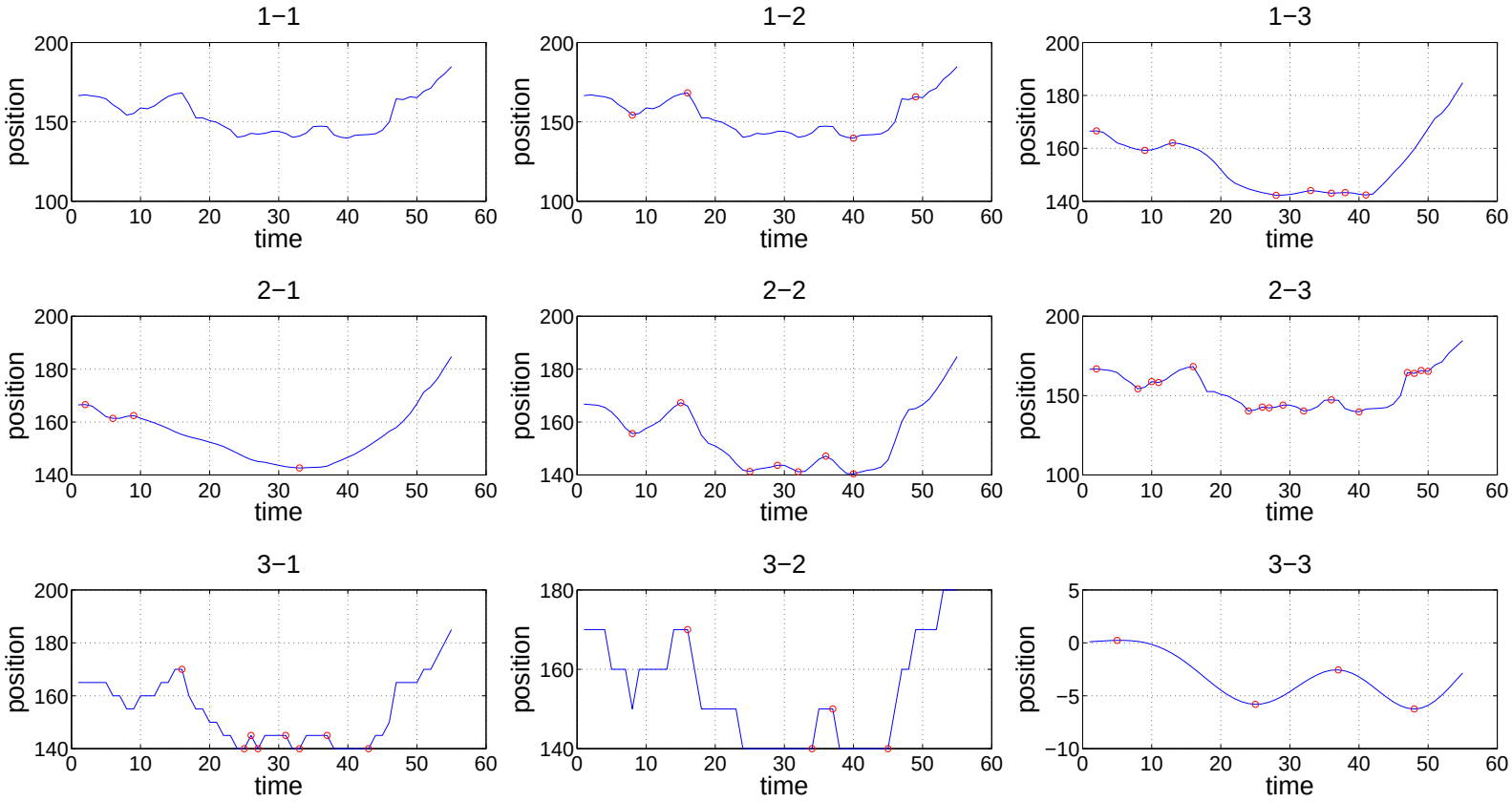

Figure 8 : Results for exemplary waveform 01. 1-1) Original waveform; 1-2) proposed hysteresis extrema seeking; 1-3)moving average with span 10 (MA-10); 2-1) MA-20; 2-2)local regression with 1 degree polynomial model (LR-1); 2-3) LR-2; 3-1)quantization with step size 5 (Q-5); 3-2) Q-10; 3-3)Fourier low pass filtering (FLPF). The red circles(○) represent the extrema found by the methods.

Experiment for extrema seeking

Outline
Introduction
Framework
Algorithm
Experiment
Conclusion

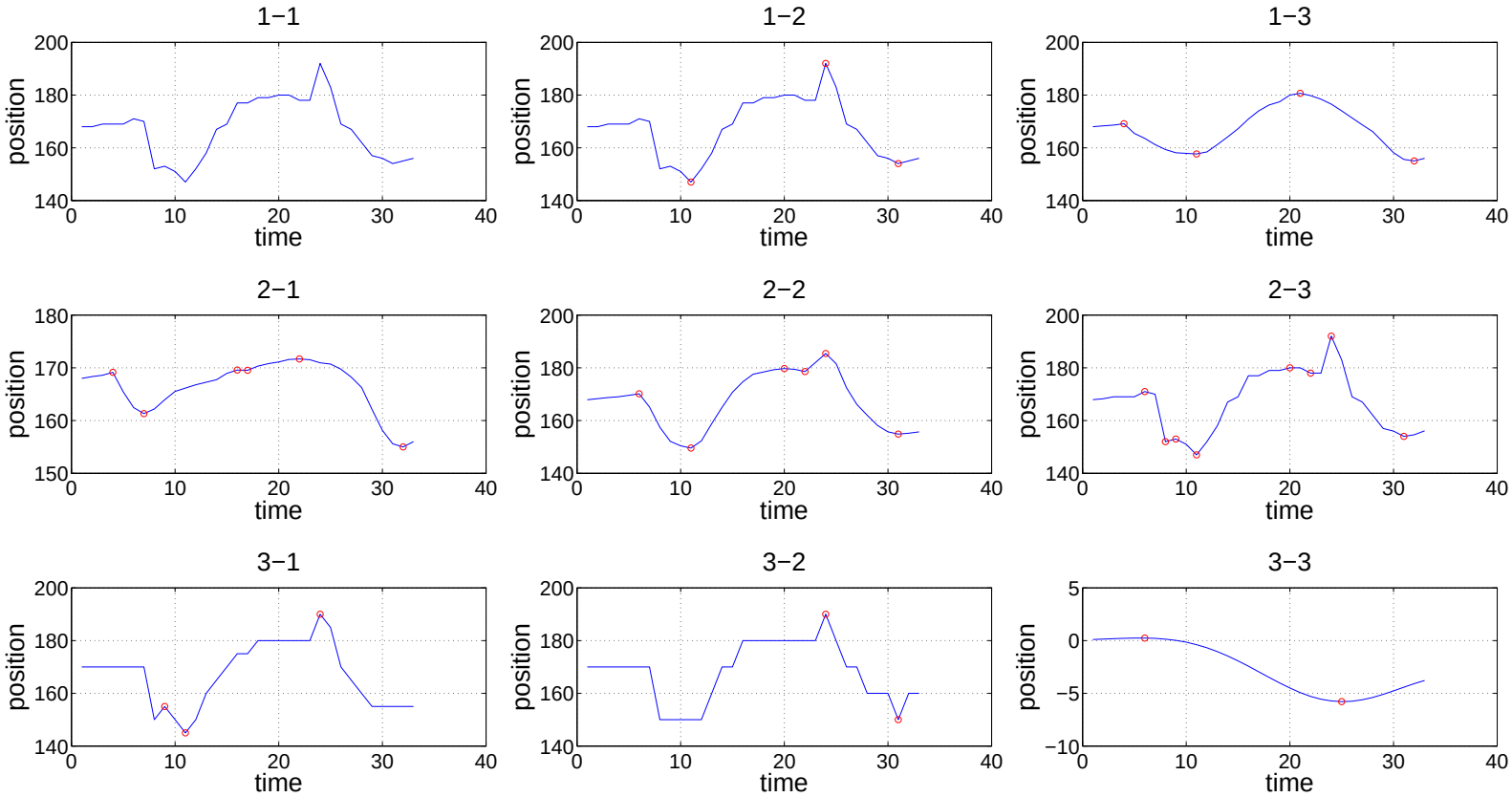

Figure 9 : Results for exemplary waveform 02. 1-1) Original waveform; 1-2) proposed hysteresis extrema seeking; 1-3)moving average with span 10 (MA-10); 2-1) MA-20; 2-2)local regression with 1 degree polynomial model (LR-1); 2-3) LR-2; 3-1)quantization with step size 5 (Q-5); 3-2) Q-10; 3-3)Fourier low pass filtering (FLPF). The red circles(○) represent the extrema found by the methods.

Experiment for extrema seeking

Outline
Introduction
Framework
Algorithm
Experiment
Conclusion

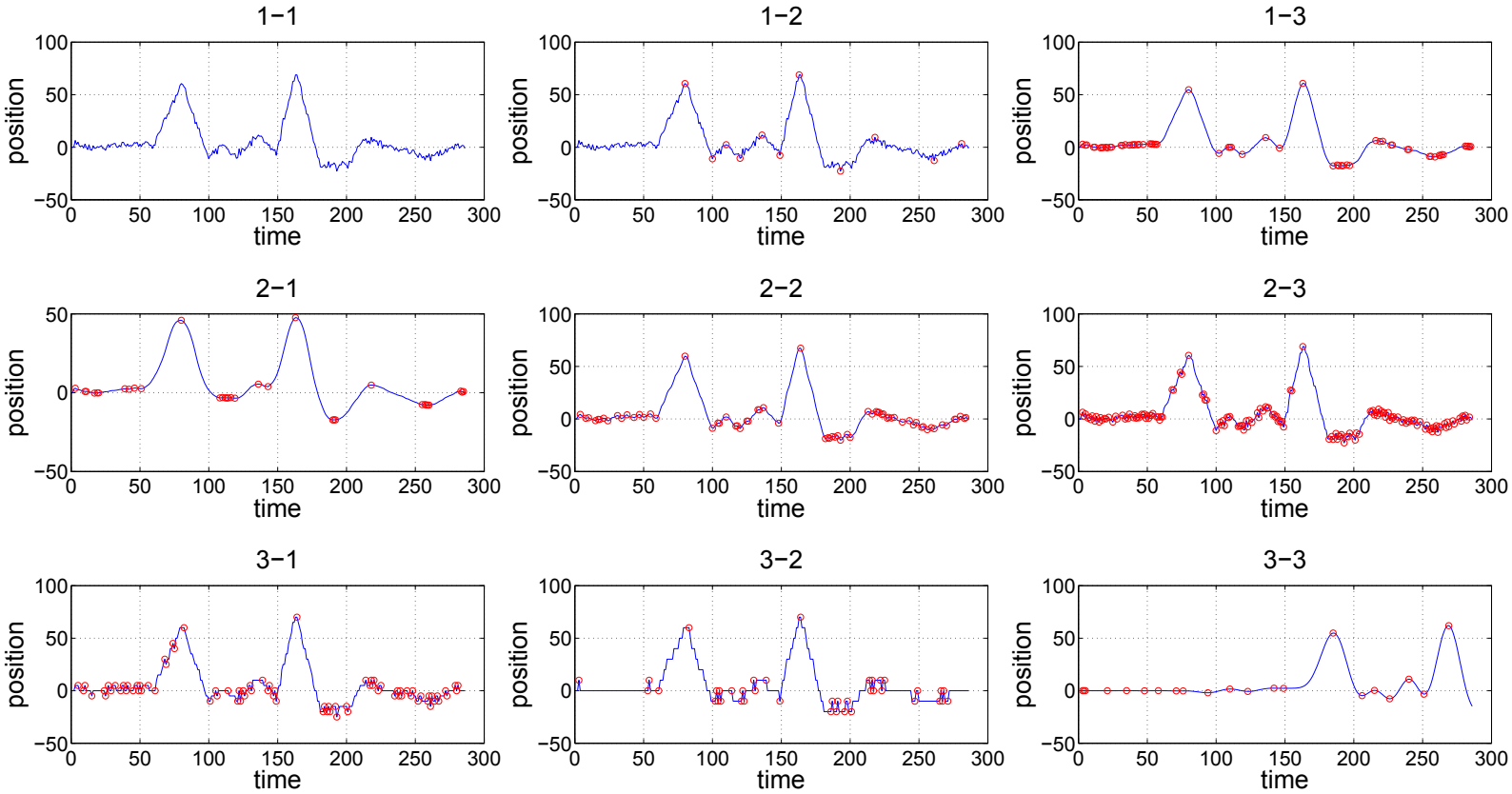

Figure 10 : Results for exemplary synthesized waveform. 1-1) Original waveform; 1-2) proposed hysteresis extrema seeking; 1-3)moving average with span 10 (MA-10); 2-1) MA-20; 2-2)local regression with 1 degree polynomial model (LR-1); 2-3) LR-2; 3-1)quantization with step size 5 (Q-5); 3-2) Q-10; 3-3)Fourier low pass filtering (FLPF). The red circles(○) represent the extrema found by the methods.

Experiment for extrema seeking

Outline

Introduction

Framework

Algorithm

Experiment

Conclusion

Table 2 : Number of Extrema

Method	Number of extrema	Method	Number of extrema
Proposed	11	LR-2	155
MA-10	63	Q-5	81
MA-20	33	Q-10	40
LR-1	67	FLPF	21

Experiment for key frame

Outline

Introduction

Framework

Algorithm

Experiment

Conclusion

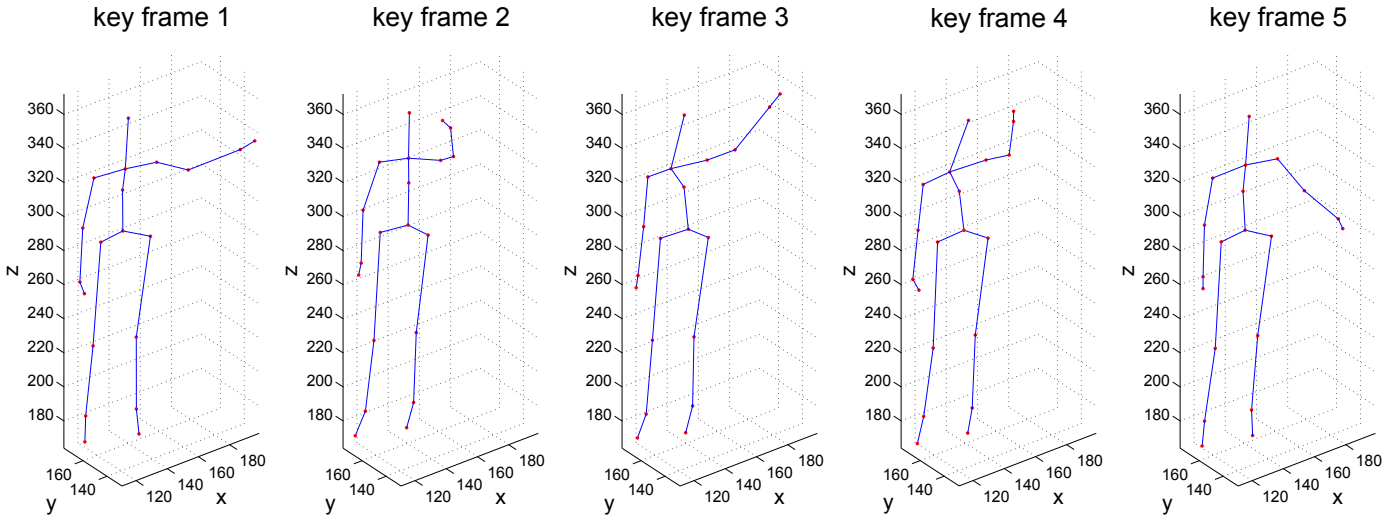

Figure 11 : Key frames extracted from action *waving*

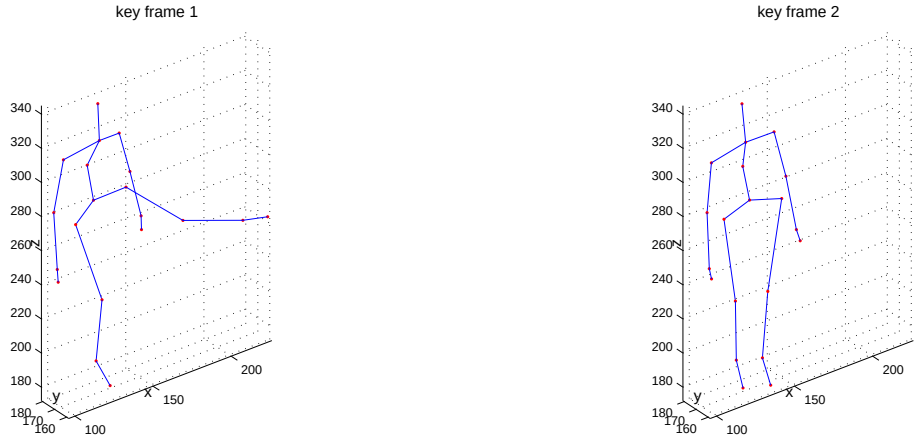

Figure 12 : Key frames extracted from action *side kick*

Conclusion

Outline

Introduction

Framework

Algorithm

Experiment

Conclusion

- A new key-frame extraction framework for RGB-D video based on motion change points.
- The hysteresis extrema seeking algorithm
  - Robust to detect extrema (motion change points)
  - Effective and easy to use with only one threshold parameter needed
- Future work
  - Locating key frames is the first stage.
  - Robust action recognition and complex action recognition.

Outline

Introduction

Framework

Algorithm

Experiment

Conclusion

Thank You

Q&A