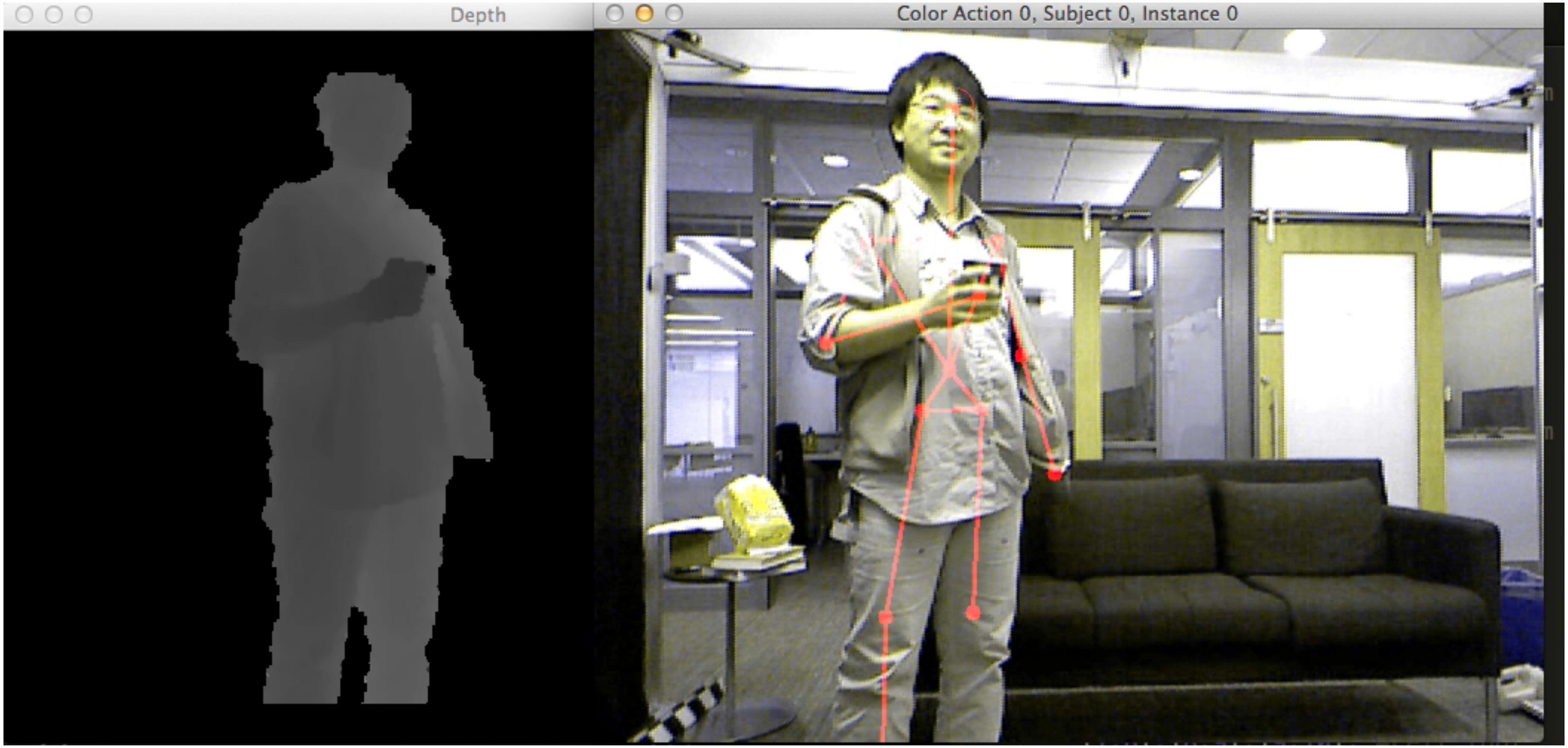
Mining Actionlet Ensemble for Action Recognition with Depth Cameras

Jiang Wang[1] Zicheng Liu[2] Ying Wu[1] Junsong Yuan[3]

[1]Northwestern University [2]Microsoft Research [3]Nanyang Technological University

(CVPR 2012)



Activity Recognition Reading Group - 4/23/2013

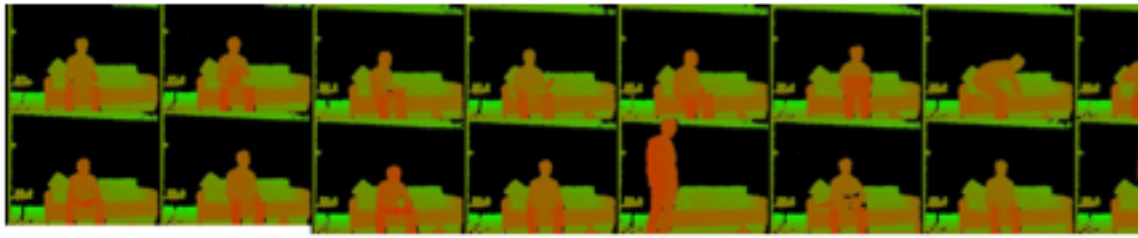
Datasets

- **MSR Action 3D**

20 Actions (e.g. high arm wave, horizontal arm wave, forward punch, high throw)

10 people, 3x each - 26 minutes total footage, 402 total actions

15 FPS, 640x480



- **MSR Daily Activity 3D**

16 Actions (e.g. drink, eat, read book, call cellphone, cheer up, play guitar, lay down)

10 people, 2x each - 320 total actions

“Living room activities”

- **CMU MoCap dataset**

5 actions (walking, marching, dribbling, walking with stiff arms, walking with wild legs)

? People, ? iterations

Only skeletons

Contributions

- Actionlet ensemble
  - (their pipeline)
- Features
  - Relative positions
  - Local Occupancy Pattern
- Temporal representation
  - Fourier temporal pyramid

Overview

1) Extract features:

   3D Local Binary Patterns ("LOP")

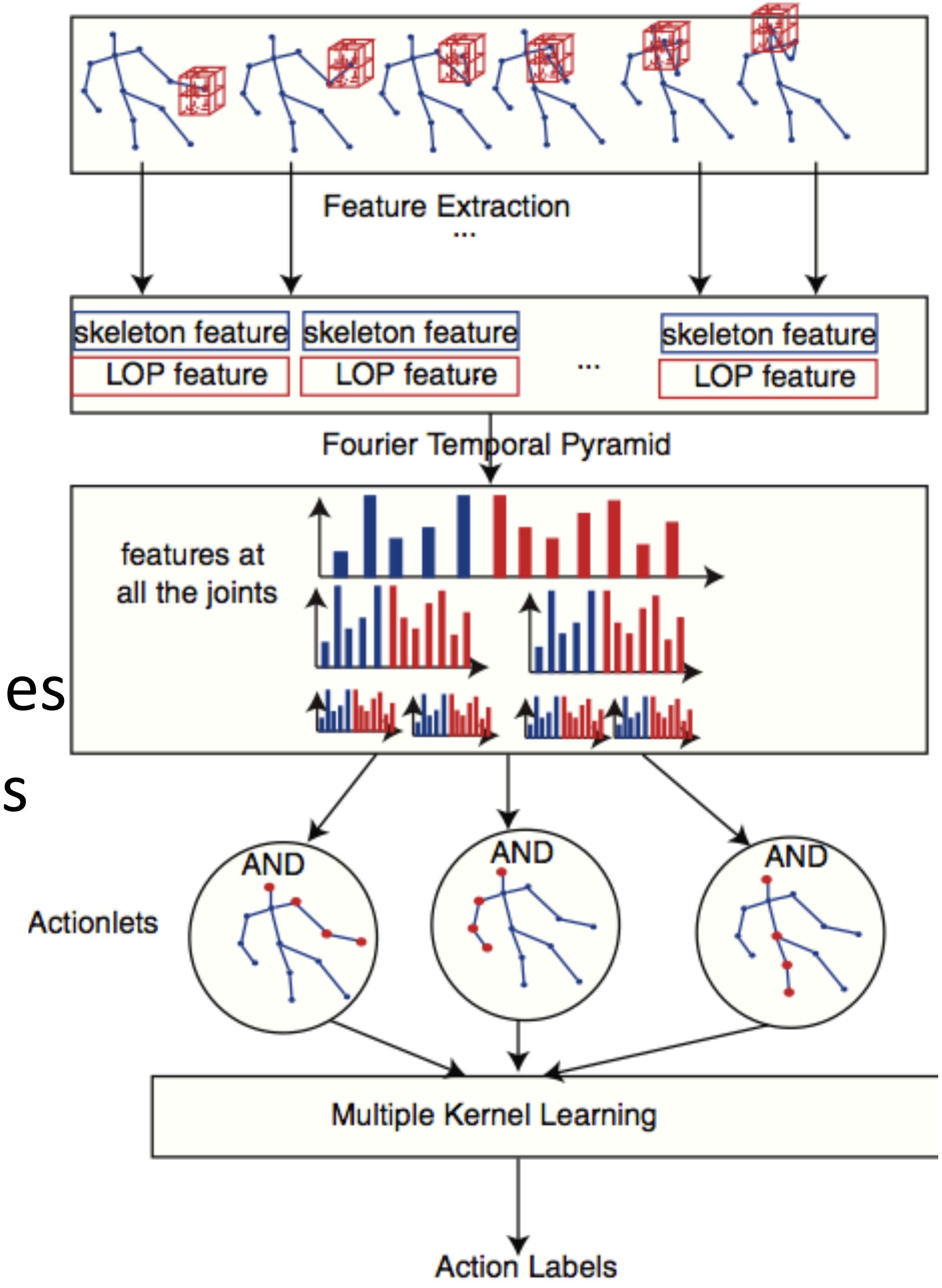
   Skeletal Features

2) Combine features

3) Look at different time scales

4) Combine *top* joint features

5) MKL classification



Features (1/2)

Invariant 3D joint positions

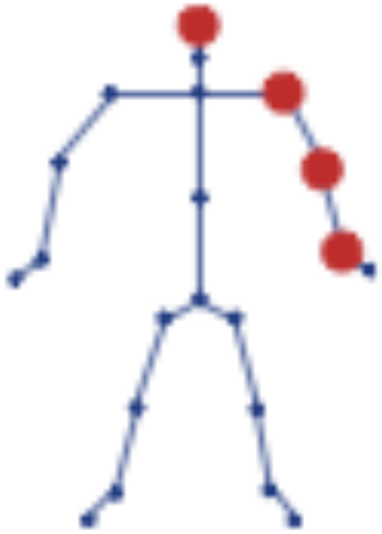
- Pairwise-relative positions

pairwise

body part feature

$$\boldsymbol{p}_{ij} = \boldsymbol{p}_i - \boldsymbol{p}_j \qquad \boldsymbol{p}_i = \{\boldsymbol{p}_{ij} | i \neq j\}$$

- Normalize
- Invariances: Translation, body size
- Robust to noise & temporal misalignment



Features (2/2)

Local Occupancy Patterns

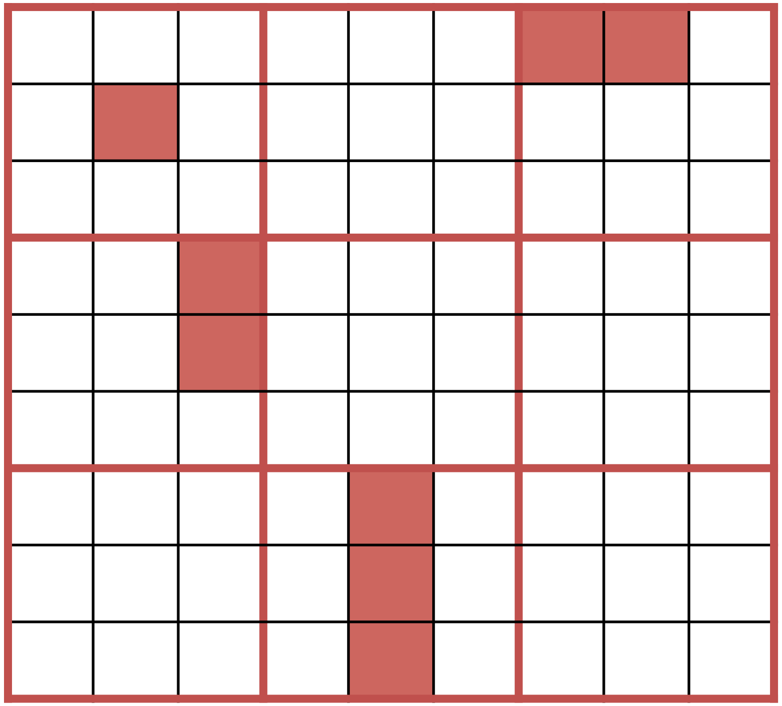
- Related Work: SIFT, STIPS, HOG, Cuboids, LBP …
- Model interactions

For each joint:

1) Partition into N_x x N_y x N_z grid.
2) Sum all entries in each bin
3) Apply sigmoid
4) Concatenate value at all bins

$$o_{xyz} = \delta\Big(\sum_{q \in \text{bin}_{xyz}} I_q \Big)$$

$$\delta(x) = \frac{1}{1+e^{-\beta x}}$$



X = [1 0 2 2 0 0 0 3 0]

Fourier Temporal Pyramid

For periodic motion, techniques like DTW produce large misalignment

Idea: recursively partition pyramid

Apply Short Time Fourier Transform

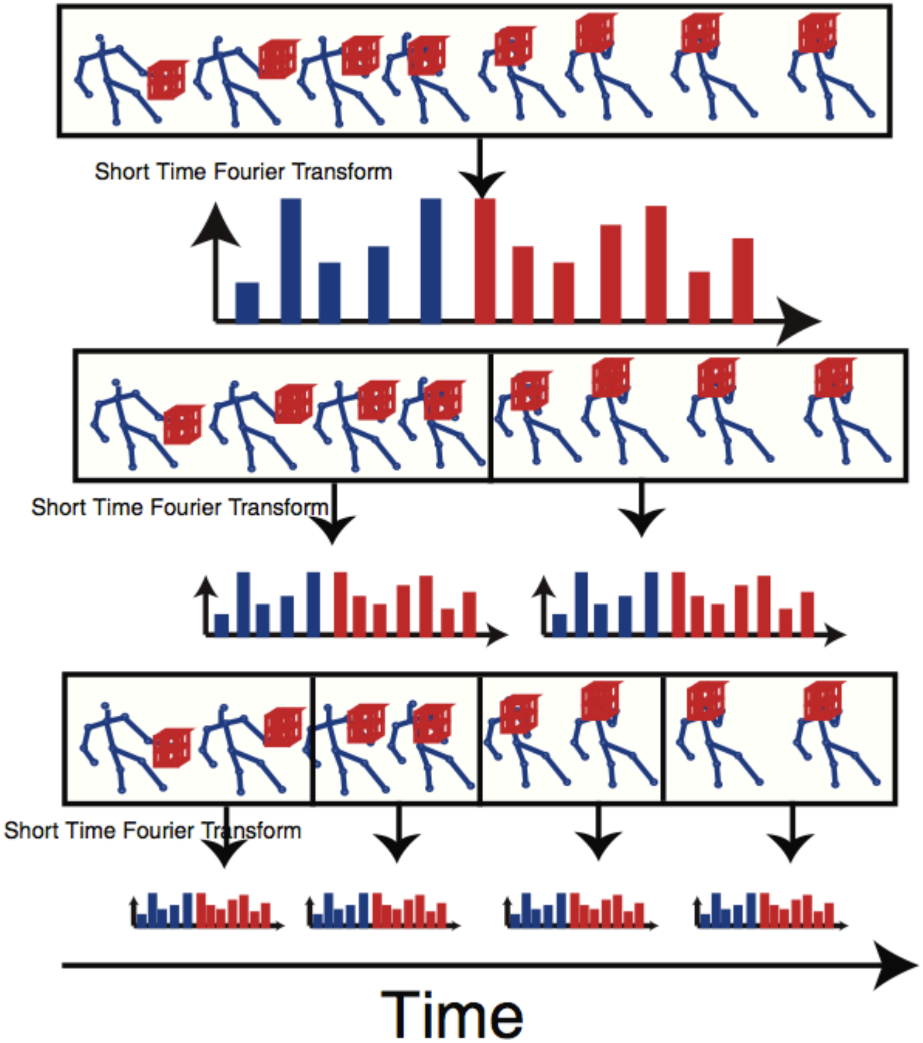
to feature set g[t] = [Pos[t], LOP[t]]

Use low frequency coeffs as features

Feature G_i: concatenate all segments

e.g. all 7 sets ->

In results: 3 levels,cutoff=1/4 length


Short Time Fourier Transform
Short Time Fourier Transform
Short Time Fourier Transform
Time

Actionlet ensemble (1/4)

Correlation of body parts is important!

actionlet := conjunctive (AND) structure on base features

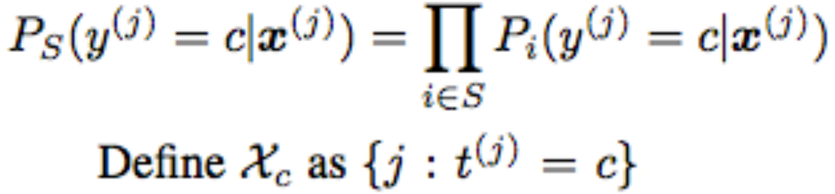
denoted by S = {1,2,…,N_j}

base feature := a fourier pyramid of one joint

Learn discriminatively which joints should be modeled together

Use AND/OR structure:

Prediction (y) is labeled (c) iff all joint features (x_j) are labeled c



$$P_S(y^{(j)} = c|\boldsymbol{x}^{(j)}) = \prod_{i \in S} P_i(y^{(j)} = c|\boldsymbol{x}^{(j)})$$

Define $\mathcal{X}_c$ as $\{j : t^{(j)} = c\}$

Actionlet ensemble (2/4)

Maximize confidence

$$\mathrm{Conf}_S = \max_{j \in \mathcal{X}_c} \log P_S(y^{(j)} = c | \boldsymbol{x}^{(j)})$$

Minimize ambiguity

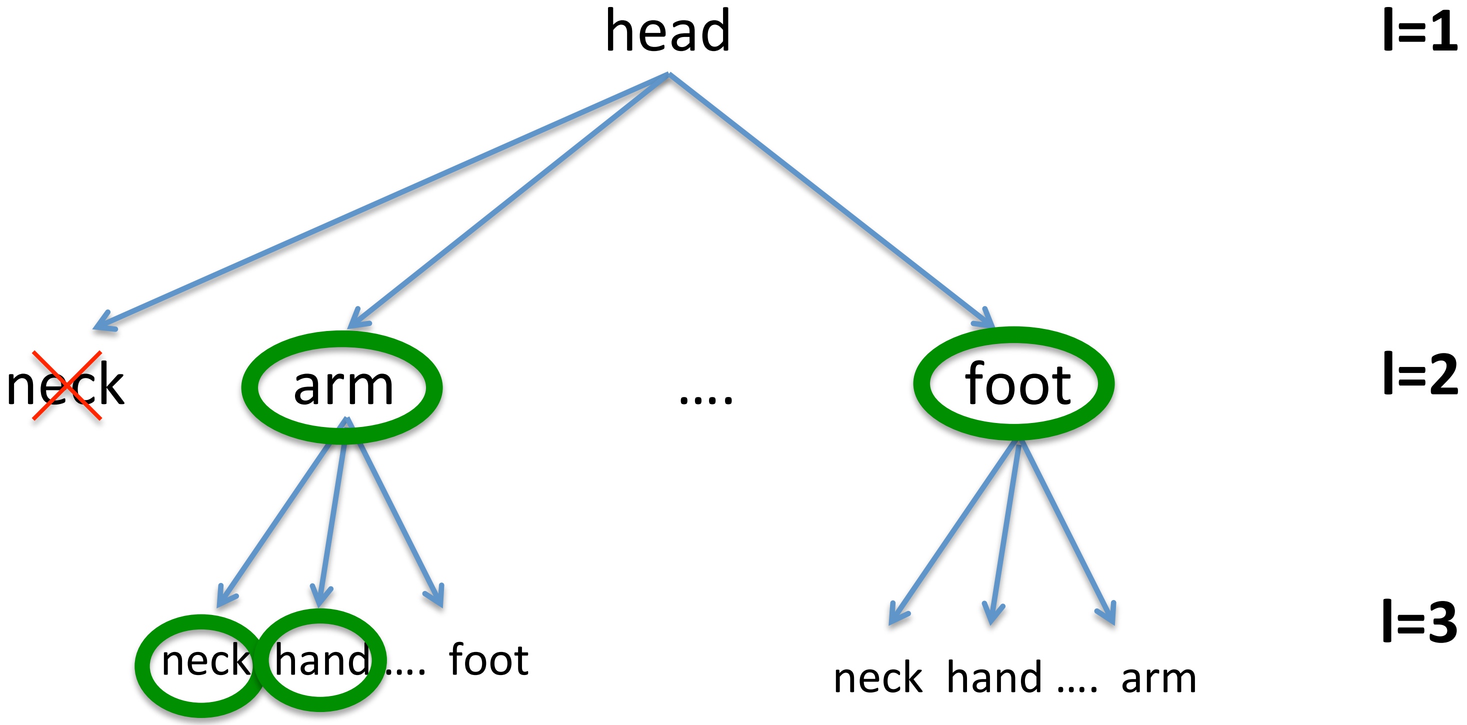
$$\mathrm{Amb}_S = \sum_{j \notin \mathcal{X}_c} \log P_S(y^{(j)} = c | \boldsymbol{x}^{(j)})$$

P_c := Discriminative actionlet pool for class c

Algorithm:

```
For c in 1…C:  // Each class
    $P_c$ <- {}     // Initialize pool
    l <- 1
    Do:
        Generate l-actionlets by adding one joint into
        each (l-1) actionlet in $P_c$
        Add l-actionlets if conf > $T_{conf}$
        l++
    Until no actionlet is added
    Remove actionlets if Ambig > $T_{amb}$ in $P_c$
Output: all actionlets that meet the criteria
```

Actionlet ensemble (3/4)



e.g. actionlets: S={head, arm, neck}
S={head, arm, hand}

Actionlet ensemble (4/4)

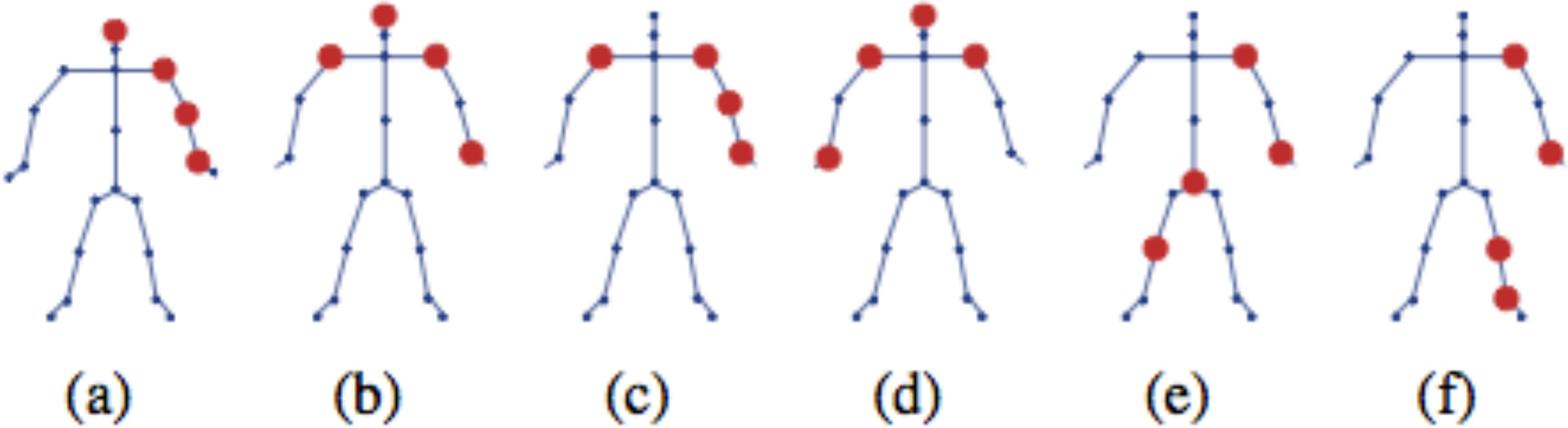


Figure 9. Examples of the mined actionlets. The joints contained in each actionlet are marked as red. (a), (b) are actionlets for "drink" (c), (d) are actionlets for "call". (e), (f) are actionlets for "walk".

Multiple Kernel Learning

- Multiclass-MKL
  - One versus all

$$f_{\text{final}}(\boldsymbol{x}, y) = \sum_{k=1}^{p} [\overset{\substack{\text{kernel}\\\text{weight}\\\downarrow}}{\beta_k} \langle \boldsymbol{w}_k, \overset{\substack{\text{kernel}\\\downarrow}}{\Phi_k}(\boldsymbol{x}, y) \rangle + b_k]$$

L1 regularizer on beta so small number of actionlets are learned

$$\Omega(\boldsymbol{\beta}) = \|\boldsymbol{\beta}\|_1^2$$

$$\min_{\boldsymbol{\beta}, \boldsymbol{w}, \boldsymbol{b}, \xi} \frac{1}{2}\Omega(\boldsymbol{\beta}) + C \sum_{i=1}^{n} \xi_i$$

$$\text{s.t. } \forall i : \xi_i = \max_{u \neq y_i} l(f_{\text{final}}(\boldsymbol{x}^{(i)}, y^{(i)}) - f_{\text{final}}(\boldsymbol{x}^{(i)}, u))$$

Solve by iteratively:

1) optimizing beta with fixed w, b with linear programming

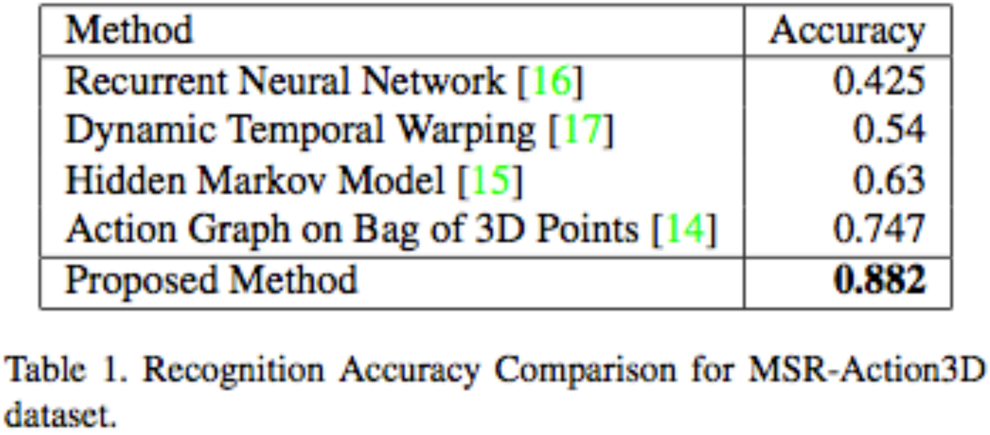
2) optimization w,b with fixed beta using generic SVM solver

MSR Action3D

Half of subjects used as training data

Errors when actions are too similar (“hand catch” and “high throw”)

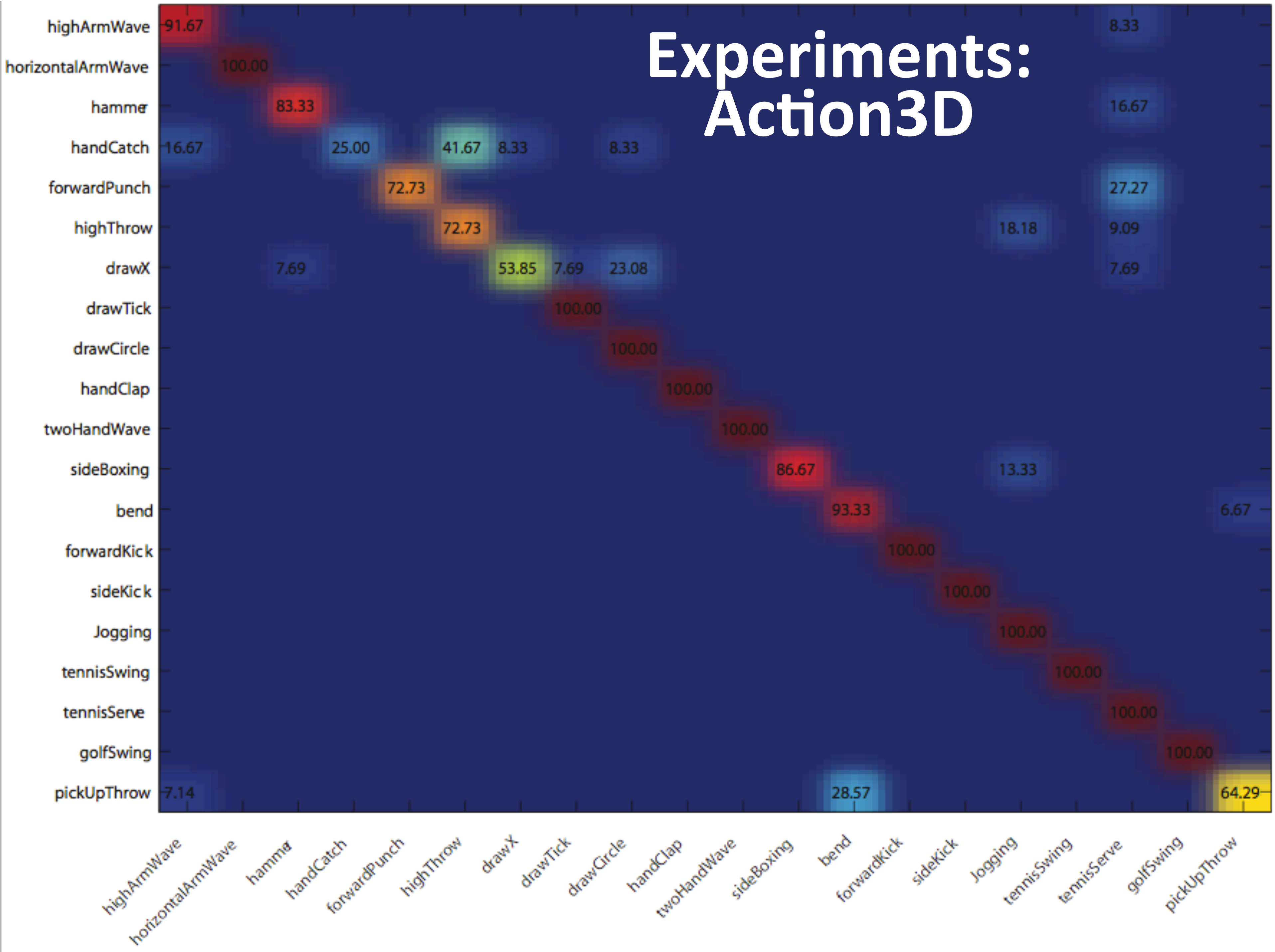
Other models thrown by temporal misalignment



Method	Accuracy
Recurrent Neural Network [16]	0.425
Dynamic Temporal Warping [17]	0.54
Hidden Markov Model [15]	0.63
Action Graph on Bag of 3D Points [14]	0.747
Proposed Method	**0.882**

Table 1. Recognition Accuracy Comparison for MSR-Action3D dataset.

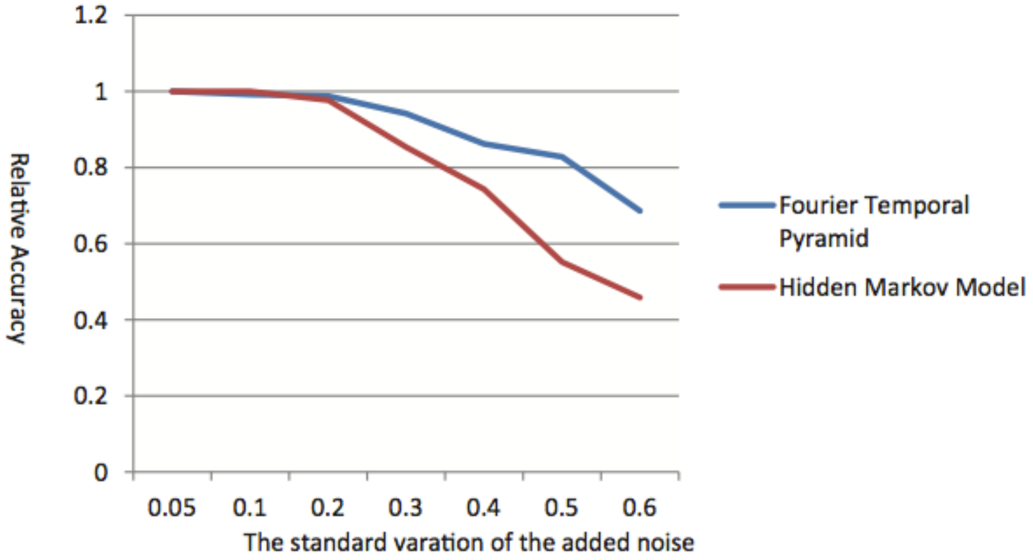
Experiments: Action3D



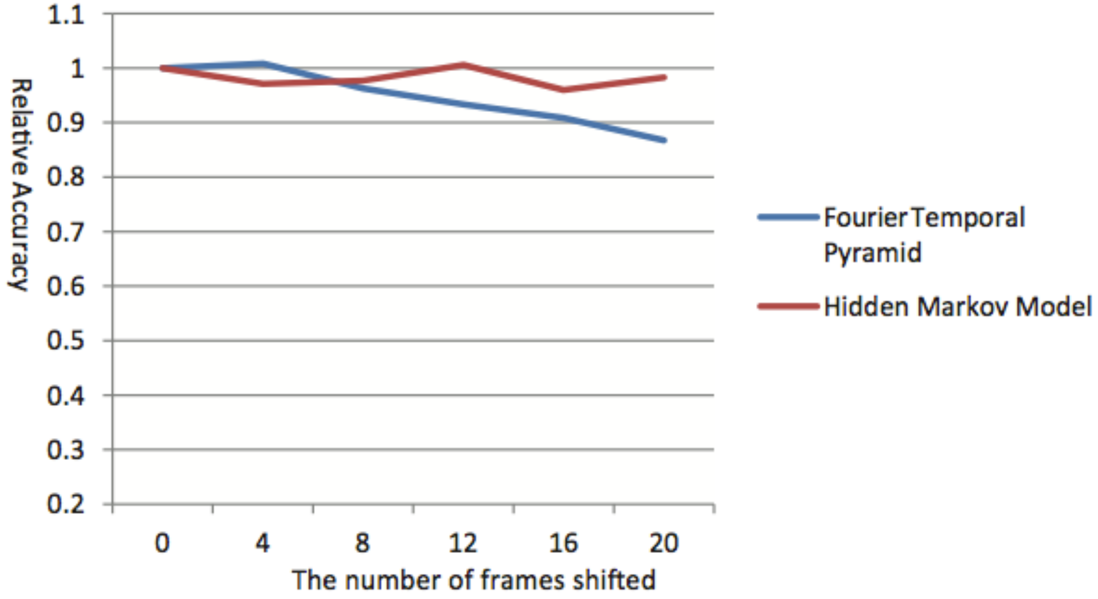
	highArmWave	horizontalArmWave	hammer	handCatch	forwardPunch	highThrow	drawX	drawTick	drawCircle	handClap	twoHandWave	sideBoxing	bend	forwardKick	sideKick	Jogging	tennisSwing	tennisServe	golfSwing	pickUpThrow
highArmWave	91.67																	8.33		
horizontalArmWave		100.00																		
hammer			83.33															16.67		
handCatch	16.67			25.00		41.67	8.33		8.33											
forwardPunch					72.73													27.27		
highThrow						72.73										18.18		9.09		
drawX			7.69				53.85	7.69	23.08									7.69		
drawTick								100.00												
drawCircle									100.00											
handClap										100.00										
twoHandWave											100.00									
sideBoxing												86.67				13.33				
bend													93.33							6.67
forwardKick														100.00						
sideKick															100.00					
Jogging																100.00				
tennisSwing																	100.00			
tennisServe																		100.00		
golfSwing																			100.00	
pickUpThrow	7.14												28.57							64.29

Sensitivity Action3D

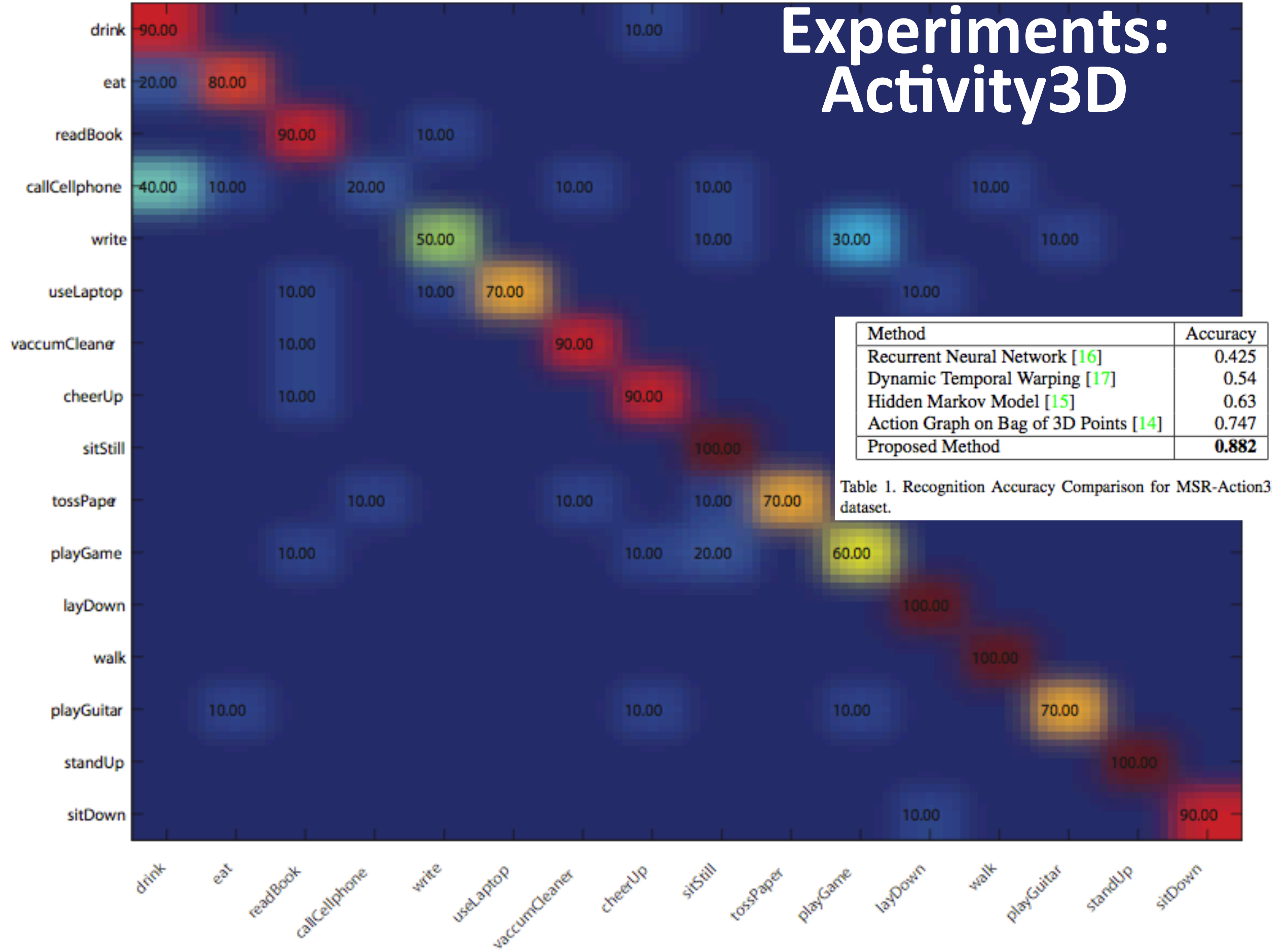
+Gaussian noise to joints



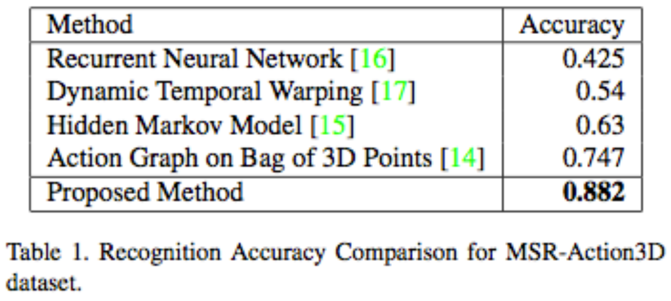
Temporal shift



Experiments: Activity3D



	drink	eat	readBook	callCellphone	write	useLaptop	vaccumCleaner	cheerUp	sitStill	tossPaper	playGame	layDown	walk	playGuitar	standUp	sitDown
drink	90.00							10.00								
eat	20.00	80.00														
readBook			90.00		10.00											
callCellphone	40.00	10.00		20.00			10.00		10.00				10.00			
write					50.00				10.00		30.00			10.00		
useLaptop			10.00		10.00	70.00						10.00				
vaccumCleaner			10.00				90.00									
cheerUp			10.00					90.00								
sitStill									100.00							
tossPaper				10.00			10.00		10.00	70.00						
playGame			10.00					10.00	20.00		60.00					
layDown												100.00				
walk													100.00			
playGuitar		10.00						10.00			10.00			70.00		
standUp															100.00	
sitDown												10.00				90.00



Method	Accuracy
Recurrent Neural Network [16]	0.425
Dynamic Temporal Warping [17]	0.54
Hidden Markov Model [15]	0.63
Action Graph on Bag of 3D Points [14]	0.747
Proposed Method	**0.882**

Table 1. Recognition Accuracy Comparison for MSR-Action3D dataset.

Experiments: DailyActivities

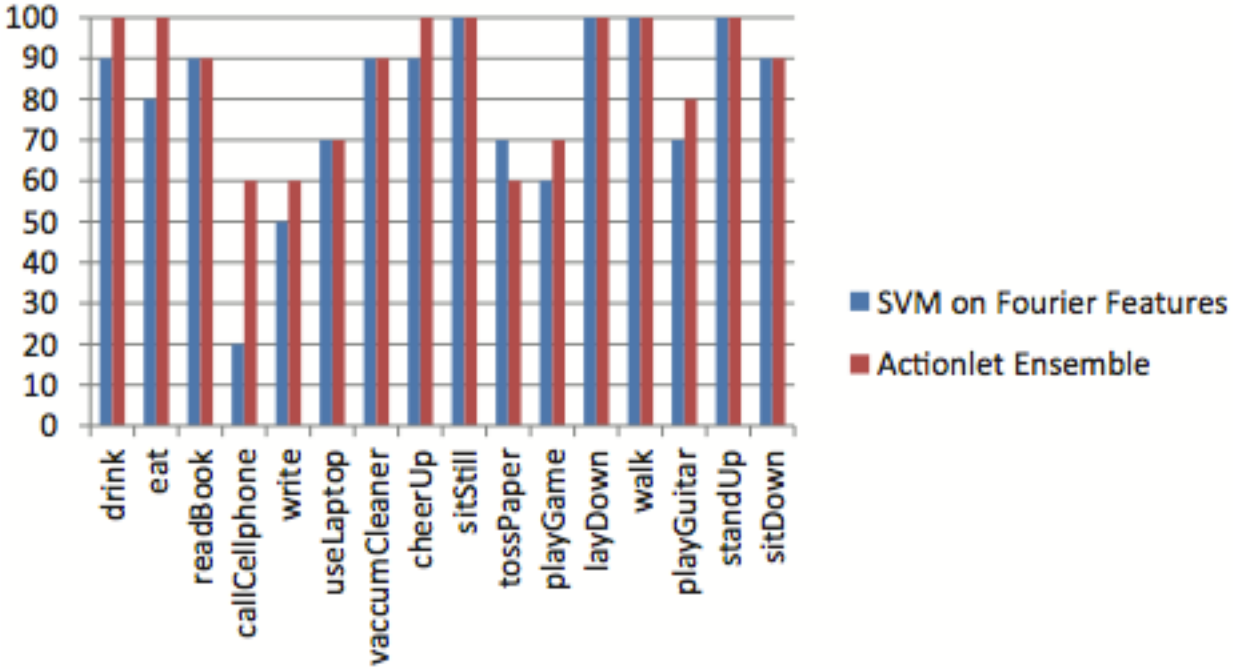
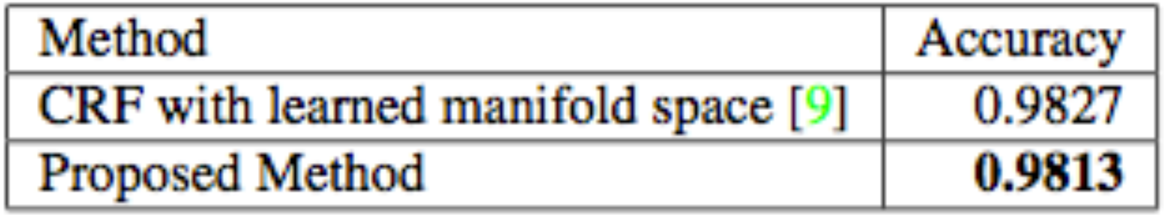


Figure 8. The comparison between the accuracy of the proposed actionlet ensemble method and that of the support vector machine on the Fourier Temporal Pyramid features.

Experiments:
CMU Mocap

Data is much cleaner than from the Kinect



Method	Accuracy
CRF with learned manifold space [9]	0.9827
Proposed Method	**0.9813**

Table 3. Recognition Accuracy Comparison for CMU MoCap dataset.