Mining Actionlet Ensemble for Action Recognition with Depth Cameras

Jiang Wang\textsuperscript{1}  Zicheng Liu\textsuperscript{2}  Ying Wu\textsuperscript{1}  Junsong Yuan\textsuperscript{3}

\textsuperscript{1}Northwestern University \textsuperscript{2}Microsoft Research \textsuperscript{3}Nanyang Technological University

(CVPR 2012)
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

\[ p_{ij} = p_i - p_j \]

\[ p_i = \{p_{ij} | i \neq j\} \]

- 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 $N_x \times N_y \times N_z$ grid.
2) Sum all entries in each bin
3) Apply sigmoid
4) Concatenate value at all bins

\[
o_{xyz} = \delta(\sum_{q \in \text{bin}_{xyz}} I_q)
\]
\[
\delta(x) = \frac{1}{1+e^{-\beta x}}
\]

\[
X = [1 \quad 0 \quad 2 \quad 2 \quad 0 \quad 0 \quad 0 \quad 0 \quad 3 \quad 0]
\]}
Fourier Temporal Pyramid

For periodic motion, techniques like DTW produce large misalignment.

Idea: recursively partition pyramid

Apply Short Time Fourier Transform to feature set \( g[t] = [\text{Pos}[t], \text{LOP}[t]] \)

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
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 $c$

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

Define $\mathcal{X}_c$ as $\{j : t^{(j)} = c\}$
Actionlet ensemble (2/4)

Maximize confidence

\[
\text{Conf}_S = \max_{j \in \mathcal{X}_c} \log P_S(y^{(j)} = c | \mathbf{x}^{(j)})
\]

Minimize ambiguity

\[
\text{Amb}_S = \sum_{j \notin \mathcal{X}_c} \log P_S(y^{(j)} = c | \mathbf{x}^{(j)})
\]

\(P_c := \) Discriminative actionlet pool for class c

**Algorithm:**

1) For \(c \in 1...C\): // Each class
2) \(P_c \leftarrow \{\}\) // Initialize pool
3) \(l \leftarrow 1\)
4) Do:
5) \(\) Generate \(l\)-actionlets by adding one joint into each \((l-1)\) actionlet in \(P_c\)
6) \(\) Add \(l\)-actionlets if \(\text{conf} > T_{\text{conf}}\)
7) \(l++\)
8) Until no actionlet is added
9) Remove actionlets if \(\text{Ambig} > T_{\text{amb}}\) in \(P_c\)

Output: all actionlets that meet the criteria
Actionlet ensemble (3/4)

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

\[
f_{\text{final}}(x, y) = \sum_{k=1}^{p} [\beta_k \langle w_k, \Phi_k(x, y) \rangle + b_k]
\]

L1 regularizer on beta so small number of actionlets are learned

\[
\begin{align*}
\min_{\beta, w, b, \xi} & \quad \frac{1}{2} \Omega(\beta) + C \sum_{i=1}^{n} \xi_i \\
\text{s.t.} & \quad \forall i : \xi_i = \max_{u \neq y_i} l(f_{\text{final}}(x^{(i)}, y^{(i)}) - f_{\text{final}}(x^{(i)}, u))
\end{align*}
\]

\[
\Omega(\beta) = \|\beta\|_1^2
\]

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

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recurrent Neural Network [16]</td>
<td>0.425</td>
</tr>
<tr>
<td>Dynamic Temporal Warping [17]</td>
<td>0.54</td>
</tr>
<tr>
<td>Hidden Markov Model [15]</td>
<td>0.63</td>
</tr>
<tr>
<td>Action Graph on Bag of 3D Points [14]</td>
<td>0.747</td>
</tr>
<tr>
<td>Proposed Method</td>
<td><strong>0.882</strong></td>
</tr>
</tbody>
</table>

Table 1. Recognition Accuracy Comparison for MSR-Action3D dataset.
Experiments: Action3D
Sensitivity
Action3D

+ Gaussian noise to joints

Temporal shift

Graph 1: Relative Accuracy vs. The standard variation of the added noise
- Blue line: Fourier Temporal Pyramid
- Red line: Hidden Markov Model

Graph 2: Relative Accuracy vs. The number of frames shifted
- Blue line: Fourier Temporal Pyramid
- Red line: Hidden Markov Model
Experiments: Activity3D

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

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRF with learned manifold space [9]</td>
<td>0.9827</td>
</tr>
<tr>
<td>Proposed Method</td>
<td>0.9813</td>
</tr>
</tbody>
</table>

Table 3. Recognition Accuracy Comparison for CMU MoCap dataset.