Kernelized Covariance for Action Recognition

Jacopo Cavazza1,2, Andrea Zunino1,2, Marco San Biagio1, Vittorio Murino1,3
1 Pattern Analysis & Computer Vision – Istituto Italiano di Tecnologia, Via Morego 30, 16163, Genova, Italy
2 Università degli Studi di Genova – Dipartimento di Ingegneria Navale, Elettrica, Elettronica e delle Telecomunicazioni, Via all’Opiu Pia, 11A, 16145, Genova, Italy
3 Università di Verona – Dipartimento di Informatica, Strada le Grazie 15, 37134, Verona, Italy
{jacopo.cavazza, andrea.zunino, marco.sanbiagio, vittorio.murino}@iit.it

Abstract

- Originally devised as an image descriptor [5], the covariance matrix is powerful in correlating skeletal joints across time for action recognition [2, 10].
- As the main limitation, covariance can only capture linear mutual relationships.
- In this work, we extend covariance to model arbitrary, non-linear relationships by recovering the applicability of the kernel trick and consequently avoiding any preliminary feature encoding of the raw data.

Experimental Results

<table>
<thead>
<tr>
<th>Method</th>
<th>MSR-Action3D</th>
<th>MSR-Daily-Activity</th>
<th>MSRSC-Kinect12</th>
<th>HDM-05</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action Graph [4]</td>
<td>79.0%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Random Occupancy Patterns [8]</td>
<td>86.5%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Actionlets [9]</td>
<td>86.2%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pose Set [7]</td>
<td>90.0%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Moving Pose [12]</td>
<td>91.7%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lie Group [6]</td>
<td>92.5%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Normal Vectors [11]</td>
<td>93.1%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kernelized-COV (proposed)</td>
<td>96.2%</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Pseudocode

1. For each action instance $X$, collect all $T$ temporal observations $x(1), \ldots, x(T)$, each one encoding the 5D coordinates of the $n$ joints.
2. For each data matrix $X$, select $h_1, \ldots, h_M$ as in Proposition 1. and compute $K(X, h)$.
3. Compute the linear operator $P$.
4. By means of $K(X, h)$ and $P$, compute $\hat{S}(k)$.

Published code available at https://www.iit.it/pavis/code/kcovr

References