Online Unsupervised Learning of the 3D Kinematic Structure of Arbitrary Rigid Bodies - Supplementary Material

Urbano Miguel Nunes and Yiannis Demiris
Personal Robotics Lab, Imperial College London, UK
{um.nunes, y.demiris}@imperial.ac.uk

6. Additional experiments

We set the same parameters for all experiments, except where mentioned otherwise, as follows: $N = 1000$, $\mu = 0.75$, $\gamma = \frac{e^{-2}}{N^2}\sum_{i,j}\tilde{d}_{ij}$, $\alpha = 50$, $\rho_{\text{thresh}} = \left\lceil \frac{2}{N}\sum_i \rho_i \right\rceil$, r as the octree resolution, $l = 3$, $\tau = 0.025$ and $\beta = 0.275$.

6.1. Influence of number of points

The input point cloud tends to be better represented by larger number of points, but the associated computational cost increases, as seen in Fig. 5. In practise, we must reach a compromise and based on Fig. 7 we can consider a lower number of points without significant performance degradation, depending on the application. For instance, in the video provided, the online and real-time estimation were performed with around 1000 and 650 points, respectively. Based on Fig. 4, the number of segments estimated is not significantly influenced by the number of points. For the tested sequences, this means that the respective point clouds are well represented by this amount of points.

6.2. Influence of μ

The $\mu \in [0, 1)$ parameter controls the contribution of the graph structure for estimating future labels, according to Eq. (10). If $\mu \to 1$, the graph structure is the dominant factor, whereas if $\mu \to 0$, future labels are more influenced by previous labels. We want to find a suitable value for μ that accounts for new information, while also considering information from previous labels. Fig. 8 shows the average results obtained, in terms of precision, recall and f-measure. We observe that higher μ values yield slightly better average results. This corroborates the fact that the AVPED can maintain current and past information of the scene. Moreover, the results of lower μ values depend significantly on the initial point cloud distribution and associated point tracking errors, which is not desirable for online and real-time applications. Based on the results from Fig. 8, we selected $\mu = 0.75$ as a balanced compromise between the contribution of the graph structure and previous labelled points.

6.3. Influence of considering l leading eigenvalues

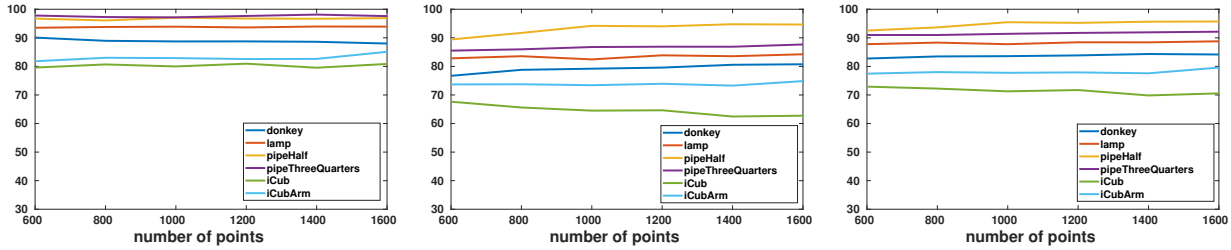
Fig. 9 shows the influence of the number of leading eigenvalues considered to build the similarity matrix, according to Eq. (8). We observe that, aside from *pipe 3/4* and *iCub* sequences, the performance is not significantly improved considering more than 3 eigenvalues. For higher number of eigenvalues considered, the average performance is only improved for the *pipe 3/4* sequence and it decreases for the *iCub* sequence.

6.4. Influence of incorporating the spectrum transformation

Due to measurement errors or faulty point tracking, noise is introduced into the AVPED. Since the AVPED is a proper distance matrix, we cast the problem of dealing with noise into embedding the AVPED in a feature space and use the resultant representation to build the similarities (as described in Section 3.3), while also incorporating the spectrum transformation into the original distance matrix $\tilde{\mathbf{D}}$ (as described in Section 3.7). Fig. 10 illustrates the benefits of this spectrum transformation process, for which there are statistically significant gains across most sequences. Also, the results are most significant when the sequences are longer and/or exhibit more noise and lost points, *e.g. iCub* and *iCubArm* sequences. Since these sequences reflect real-world scenarios more accurately, we advocate for the incorporation of spectrum transformation into AVPED.

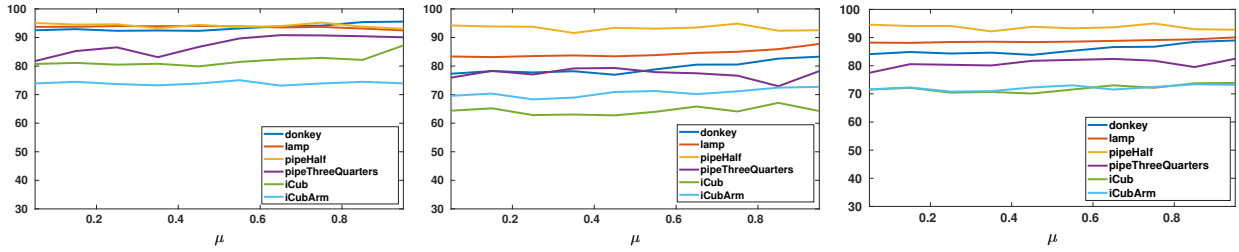
6.5. Consistency analysis in sequential estimation

The proposed method provides sequential consistency in two ways: by construction from Eq. (2), and by clustering via *label spreading*. An analysis of the number of segments estimated throughout the *iCub* and *lamp* sequences compared to clustering via spectral clustering is provided in Fig. 11. It can be seen that the *label spreading* provides additional consistency to the overall method's performance. In particular, the method's consistency is empirically robust to neighbouring values of $\mu = 0.75$.



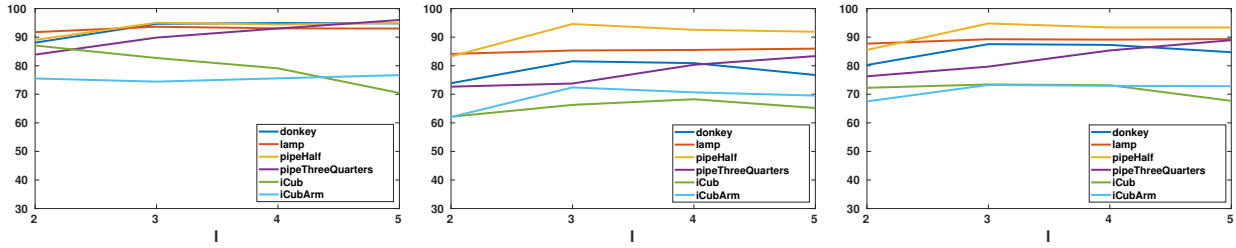
(a) Average precision (%) (b) Average recall (%) (c) Average f-measure (%)

Figure 7: Average (a) precision, (b) recall and (c) f-measure as a function of number of points considered.



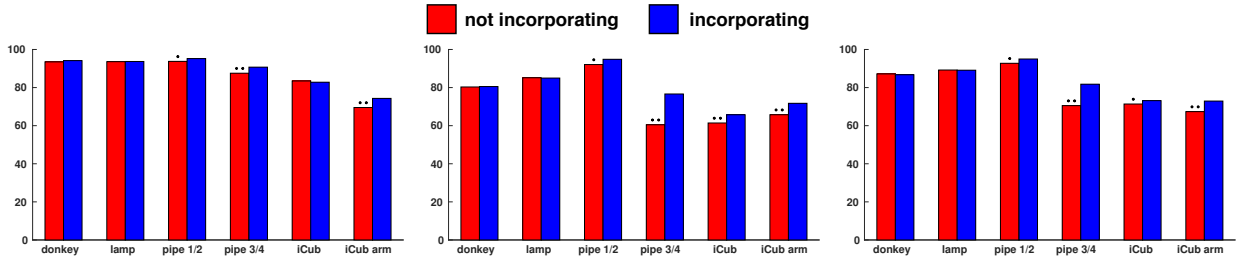
(a) Average precision (%) (b) Average recall (%) (c) Average f-measure (%)

Figure 8: Average (a) precision, (b) recall and (c) f-measure as a function of μ parameter.



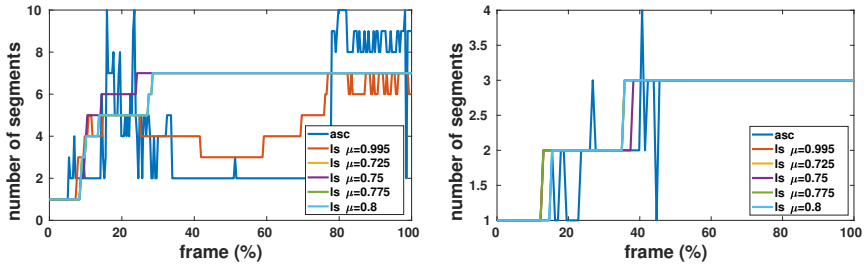
(a) Average precision (%) (b) Average recall (%) (c) Average f-measure (%)

Figure 9: Average (a) precision, (b) recall and (c) f-measure as a function of considering l leading eigenvalues, according to Eq. (8).



(a) Average precision (%) (b) Average recall (%) (c) Average f-measure (%)

Figure 10: Average (a) precision, (b) recall and (c) f-measure as a function of incorporating the spectrum transformation. * and ** represent statistical significance with p-value<0.05 and p-value<0.005, respectively.



(a) *iCub* (b) *lamp*

Figure 11: Number of segments estimated using *label spreading* (ls) and adaptive spectral clustering (asc) over (a) *iCub* and (b) *lamp* sequences. We see that the number of segments estimated by applying spectral clustering at each frame is not very consistent.