Rank estimation of trajectory matrix in motion segmentation

L. Zappella, X. Lladó and J. Salvi

A novel technique for estimating the rank of the trajectory matrix in the local subspace affinity (LSA) motion segmentation framework is presented. This new rank estimation is based on the relationship between the estimated rank of the trajectory matrix and the affinity matrix built with LSA. The result is an enhanced model selection technique for trajectory matrix rank estimation by which it is possible to automate LSA, without requiring any *a priori* knowledge, and to improve the final segmentation.

Introduction: Motion segmentation is an essential building block for many computer vision based applications. One of the most promising and newest techniques is the local subspace affinity (LSA) [1, 2]. LSA is a framework for trajectories motion segmentation under affine projection which is able to deal with different kinds of motion: rigid, non-rigid and articulated. LSA is based on the following steps: 1. given a video sequence, build a trajectory matrix $\boldsymbol{W}_{2f\times p}$, where f is the number of video frames and p is the number of tracked feature points; 2. estimate the rank r of $\boldsymbol{W}_{2f\times p}$; 3. project the trajectories onto a new space of dimension r; 4. build a trajectory affinity matrix $\boldsymbol{A}_r$ as the inverse of the distances between the subspaces generated by each trajectory in the new space; 5. cluster $\boldsymbol{A}_r$ providing the final motion segmentation.

One of the critical steps of LSA is the $\boldsymbol{W}_{2f\times p}$ rank estimation. In the original version of LSA [1] this estimation is done by a model selection (MS) technique:

$$r_k = argmin_r \frac{\lambda_{r+1}^2}{\sum_{i=1}^{r} \lambda_i^2} + kr \tag{1}$$

λ_n being the nth singular value of $\boldsymbol{W}_{2f\times p}$, and k a parameter that depends on the noise of the tracked point positions. The higher the noise level, the greater k should be [1]. However, knowing in advance the noise level is a rather big assumption and it is an obstacle that prevents the use of LSA without a tuning process. On the other hand, using a constant k or an improperly tuned one leads to poor motion segmentation results. This was also pointed out by Tron and Vidal [3] who, in their implementation of LSA avoided the MS and preferred to fix the new space size to $4n$, where n is the number of motions in the video sequence. Unfortunately, doing so two new assumptions are introduced: rigid motion (the theoretical maximum rank of $\boldsymbol{W}_{2f\times p}$ for a rigid motion is 4) and knowledge of the number of motions n. In this Letter we present a new automatic rank estimation technique for trajectory matrices that overcomes these limitations and provides a more accurate rank estimation and therefore a better motion segmentation.

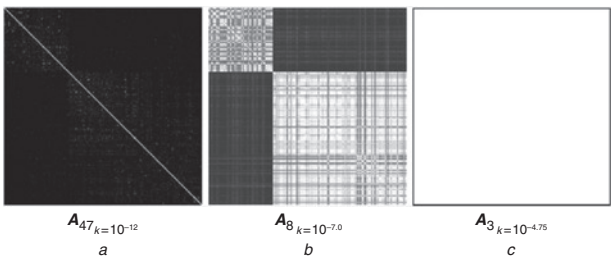

Fig. 1 *Affinity matrices of a real sequence computed with different k values (black is minimum, white is maximum)*

Enhanced model selection: Our enhanced model selection (EMS) technique tries to find automatically the correct k value for any given sequence. The key of EMS is the relationship between the MS estimation and the computed affinity matrix. When k is too small compared to the noise level, MS overestimates the rank r_k leading to an $\boldsymbol{A}_{r_k}$ where every trajectory is unlikely to be related to any other (Fig. 1*a*). On the contrary, when k is too high compared to the noise level, MS underestimates r_k, leading to an $\boldsymbol{A}_{r_k}$ where every trajectory is strongly related to all the others (Fig. 1*c*). Finally, when k is correctly tuned MS provides an accurate rank estimation leading to an $\boldsymbol{A}_{r_k}$ which contains enough information for a successful clustering segmentation (Fig. 1*b*). Therefore, $\boldsymbol{A}_r$ seems to carry some information about the accuracy of the previously estimated rank r. Hence, without assuming any knowledge about the number and the types of motion, we can iterate the rank estimation and the affinity matrix computation until a 'good' affinity matrix is obtained.

Ideally, if there are at least two motions in the sequence, an affinity matrix should have only two values: minimum and maximum affinity. In practice, owing to noise and dependent motions, a good affinity matrix is not binary, but it does have two modes plus a certain amount of in-between values. To evaluate the quality of the affinity matrix we analysed the trend of different statistical parameters, going from overestimation to underestimation of the rank of $\boldsymbol{W}_{2f\times p}$. It emerged that entropy is defined as in the following equation:

$$E(\boldsymbol{A}_{r_k}) = -\sum(\boldsymbol{I}_{r_k}\log_2(\boldsymbol{I}_{r_k})) \tag{2}$$

where $\boldsymbol{I}_{r_k}$ contains the histogram counts of the affinity matrix $\boldsymbol{A}_{r_k}$, can be used to describe the goodness of $\boldsymbol{A}_{r_k}$. In fact, when $\boldsymbol{A}_{r_k}$ histogram contains only low or high values the entropy is low, while when $\boldsymbol{A}_{r_k}$ histogram tends to have a bimodal distribution the entropy increases. Hence, the aim is to find the k that leads to the affinity matrix with maximum entropy.

However, the amount of computations in order to find the exact position of the maximum entropy is considerable. Therefore, to speed up the process we exploited a property of the entropy trend. Given k_1 and k_2, with $k_1 < k_2$, they lead to r_{k_1} and r_{k_2}, where $r_{k_1} \geq r_{k_2}$. This means that the space size onto which the trajectories are projected becomes smaller as k increases. Because $\boldsymbol{A}_{r_k}(x, y)$ of any given pair of trajectories x and y is the inverse of the distance of the generated subspaces, then $\boldsymbol{A}_{r_{k_1}}(x, y) \leq \boldsymbol{A}_{r_{k_2}}(x, y)$. Iterating this process with increasing k_i generates at the beginning $\boldsymbol{A}_{r_{k_i}}$ matrices with very low values. As k_i increases, r_{k_i} tends towards the real rank of $\boldsymbol{W}_{2f\times p}$, hence the $\boldsymbol{A}_{r_{k_i}}$ matrices histogram starts to become bimodal. However, when r_{k_i} becomes smaller than the real rank of $\boldsymbol{W}_{2f\times p}$, then the $\boldsymbol{A}_{r_{ki}}$ matrices histogram starts to become unimodal towards the maximum value. In other words, the function generated by the entropy computed on all the $\boldsymbol{A}_{r_k}$ matrices has only one global maximum and no local maxima nor minima. Hence, it is possible to compute only some $\boldsymbol{A}_{r_k}$ matrices, and their entropy value, and interpolate the entropy trend in order to have an accurate approximation of where the maximum entropy is, drastically reducing the amount of calculations.

Results: To evaluate EMS we compared the results obtained using LSA with MS (implementation by Tron and Vidal available at http://www.vision.jhu.edu) and our implementation of LSA with EMS (available at http://eia.udg.edu/~zappella). Both algorithms provide the final segmentation applying spectral clustering to the affinity matrix as suggested in [1]. We used the Hopkins155 database [3], which is a reference benchmark database for motion segmentation composed of 156 real video sequences: 120 with two motions and 36 with three motions.

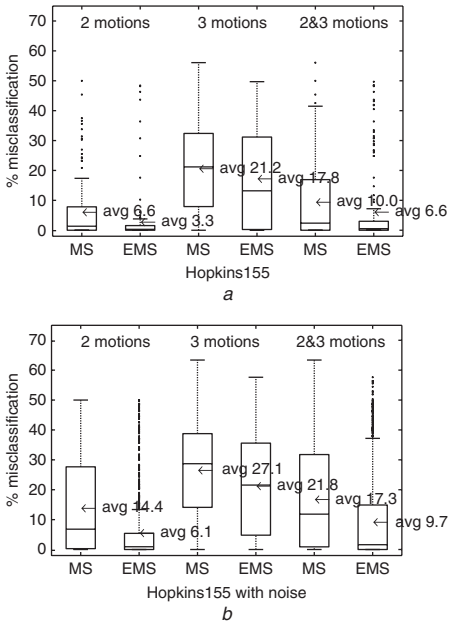

Fig. 2 *Boxplots of misclassification rates of LSA with MS and LSA with EMS*

Misclassification percentages are shown in the boxplots of Fig. 2*a*. It should be noted that EMS always has a lower average misclassification. Both algorithms have more problems to deal with three motions but also in this case EMS performs better than MS. Moreover, the lower and upper quartile ranges with EMS are always more compact. This is notable especially if it is considered that for MS we did a tuning process and we are presenting here the lowest average misclassification (obtained with $k = 10^{-7}$), whereas with EMS we did not have to do any tuning or pre-computation. These results prove that EMS provides a better rank estimation and it does so in an automatic fashion.

The Hopkins155 database contains three types of sequences: checkboards, traffic and articulated/non-rigid sequences. The main group is the checkboard which contains 104 videos. This means that among the sequences it is likely that the type and the amount of noise does not change much as most of the sequences are taken in the same environment. To test EMS with different noise levels we created another six databases derived from the Hopkins155 adding random Gaussian noise, with standard deviation of 0.5, 1, 1.5, 2, 2.5 and 3 pixels, to the tracked point positions. The original database plus the six derived from it compose a bigger database with 1092 video sequences. We compared again LSA with MS using $k = 10^{-7}$ and LSA with EMS. The misclassification percentages of the sequences with added noise are shown in the boxplots of Fig. 2*b*. As before, LSA with EMS has lower average misclassification and more compact quartile ranges. As expected, the increment of the misclassification (from Figs. 2*a* to *b*) is greater with MS than with EMS. For two and three motions, the MS misclassification increment is more than double that of the EMS one: 7.3 against 3.1%.

Conclusions: A novel EMS technique for the estimation of the trajectory matrix rank is presented. The results confirmed that EMS provides better rank estimation, leading to a more accurate motion segmentation. Moreover, while MS requires some tuning related to the noise level, EMS is able to adapt automatically to different noise levels without any *a priori* knowledge, releasing LSA from one of its biggest assumptions. In fact, until now, LSA required either knowledge about the amount of noise in order to tune the MS [1, 2], or knowledge about the number and types of motion [3].

Acknowledgments: This work has been supported by the Spanish Ministry of Science project DPI2007-66796-C03-02. L. Zappella is supported by the Catalan government scholarship 2007FI_A 00765. We thank R. Tron and R. Vidal for sharing the database and the code.

28 January 2009
doi: 10.1049/el.2009.0254

L. Zappella, X. Lladó and J. Salvi (*Institute of Informatics and Applications, University of Girona, Girona, Spain*)

E-mail: zappella@eia.udg.edu

References

1 Yan, J., and Pollefeys, M.: 'A general framework for motion segmentation: independent, articulated, rigid, non-rigid, degenerate and non-degenerate'. European Conf. on Computer Vision, Austria, May 2006, pp. 94–106
2 Yan, J., and Pollefeys, M.: 'A factorization-based approach for articulated nonrigid shape, motion and kinematic chain recovery from video', *IEEE Trans. Pattern Anal. Mach. Intell.* 2008, **30**, (5), pp. 865–877
3 Tron, R., and Vidal, R.: 'A benchmark for the comparison of 3-D motion segmentation algorithms'. IEEE Conf. on Computer Vision and Pattern Recognition, Minneapolis, MN, USA, June 2007, pp. 1–8