Multiple Model Fitting By Evolutionary Dynamics

Michael Donoser, Martin Hirzer and Dieter Schmalstieg
Institute for Computer Graphics and Vision
Graz University of Technology, Austria
Email: {donoser,hirzer,schmalstieg}@icg.tugraz.at

***Abstract*—We propose a novel multiple model fitting method based on outlier insensitive evolutionary dynamics, fulfilling several important requirements. Our method automatically identifies a unspecified number of models and is robust to noise and outliers in the data. Furthermore, we are able to handle overlapping models, by allowing that data points are assigned to more than one model. This is implicitly handled during model fitting and not as a post-processing step. Gross outliers are directly identified, by letting some points unassigned. We also introduce a technique, considering nearest neighbor analysis, to significantly reduce computation time, while maintaining model fitting accuracy. We show experiments on real-world and synthetic data, achieving accurate model fitting results also demonstrating an application of plane fitting on a consumer hardware providing RGB-D video streams.**

I. Introduction

Robustly fitting models to data plays an important role in computer vision, since observations and measurements are usually quite noisy and are mostly contaminated with a large number of outliers. Since outlier rates beyond 50% are prevalent in many computer vision applications, robust statistical approaches like RANSAC have to be applied. In many applications more than a single model has to be fitted to the data, which further complicates the task. To robustly fit multiple models to outlier corrupted data, methods in general have to cope with four different types of data points: (a) inliers to a single model, (b) inliers to multiple models (if models overlap), (c) pseudo outliers, i. e. outliers to the current model but inliers to other models and (d) gross outliers, not belonging to any model. These different types of data points pose specific requirements on a multiple model fitting procedure.

First, since the number of models is a-priori unknown, the method has to automatically identify this number. Second, the method has to be robust to noise and has to cope with all aforementioned types of outliers. Third, data points belonging to several models should not adversely influence the model fitting accuracy. Fourth, each model should be assigned with an intuitive measure of model fitting quality, to ease rejection of low-quality models. Fifth, the method should be able to return unassigned data-points, i. e. gross outliers should be implicitly identified. In this paper, we present a novel method for multiple model fitting which addresses all aforementioned requirements. Figure 1 illustrates such a desired result for the task of line fitting. Related work in this field, partly addresses some of the aforementioned points as it is discussed next.

The most common approach for single model fitting is RANSAC [1], which is successfully applied in several computer vision applications. RANSAC randomly selects minimal sample sets to estimate model hypotheses. For each hypothesis, a so-called consensus set is calculated, which includes all data points that fit to the model according to an a-priori specified inlier threshold [2]. The hypothesis having the largest consensus set is finally used to estimate the model parameters in a least-squares manner. RANSAC only fits a single model to the data. A straightforward solution for obtaining several models with RANSAC would be to apply it sequentially, by simply removing already assigned data points. Obviously, inaccurate estimates of the initial models adversely affect the later model fitting accuracy and such an approach is not optimal. This issue was addressed in [3], where several models are fitted in parallel (multi-RANSAC). Nevertheless, this approach still fails when models are strongly overlapping and the number of models to obtain has to be defined in advance. In [4] RANSAC is tuned to the specific task of image matching (PROSAC), where the random sampling of the hypotheses is based on drawing progressively larger sets of top ranked correspondences obtained from feature matching. Although improving speed, such an approach requires a measure of data point quality, which is not easily obtainable for tasks like plane fitting.

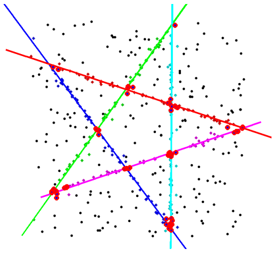

Fig. 1: Illustration of robust multiple model fitting. Our method automatically identifies the correct number of models (5) and the assignment of data points to these models. Different models are shown in different colors and multi-assigned data points are highlighted in bold (where lines cross).

The second group of algorithms in this field is based on the Hough transform concept. Here, the model parameter space is binned and minimal sample sets are randomly generated. The corresponding model parameters are inserted into a parameter histogram. The models are then identified as local maxima within the parameter space, which enables much more intuitive extraction of multiple models [5]. Nevertheless, identifying local maxima is a complex task and the accuracy is limited.

A third group of algorithms in this field focuses on solving the multiple model fitting problem by energy minimization approaches [6], [7]. Such methods have the advantage that they provide some guarantee on solution optimality, but nevertheless have high computational complexity because of their implicit complex problem formulation and mostly some unintuitive regularization parameters have to be tuned.

Available model fitting approaches can furthermore be distinguished concerning the internal representation of the data, where mainly three different types of spaces are considered: (a) *parameter*, (b) *residual* and (c) *conceptual*.

The most common approach, as considered in [3], [5] is to directly use the parameter space, where the distribution of residuals per hypothesis is analyzed. Since such approaches neglect equivalences between the models, in [8] a change of perspective was suggested. There authors proposed to analyze the similarities between the residuals of each data point to all hypothesized models. This idea is based on the fact that data point residuals cluster together to true models, if the model hypotheses are calculated in a random manner. However mode finding in this residual space is still quite difficult as shown in [3]. The same residual representation was used in [9], where a statistical learning approach for the task of model fitting was introduced. A novel kernel (Ordered Residual Kernel) for comparing data points was described, where mainly similarities between the nearest neighbor sets, concerning the residual representation, are calculated. The proposed method reduces the number of parameters to be fixed by using standard statistical approaches applicable for such Mercer kernels. Nevertheless, the method includes an error-prone outlier rejection step and requires a subsequent model-merging step since it provides highly over-segmented model fitting results. A third representation was used in [10], where authors follow the idea of residual analysis as in [8], but in contrast switch to a conceptual representation, where each data point is represented by a binary vector showing the consensus set to the hypothesized models. This allows similarity analysis using set overlap criterions like Jaccard distances, and therefore the method was denoted as J-Linkage. Nevertheless, J-Linkage does not allow overlapping models, i. e. makes a unique assignment of data points to models. Furthermore, all outliers appear as small clusters and a cumbersome rejection strategy has to be applied to reject invalid models.

We propose a novel method which in analogy to [10], [8] aims at clustering data points by considering the similarity of their residuals to randomly selected hypotheses. The core part of our method are evolutionary dynamics for identifying data-points that have a high internal coherency concerning the similarity of residuals to the hypothesized models. Our main contribution is a novel formulation, which allows the assignment of data points to several, potentially overlapping models in an implicit manner. This additionally enables to automatically identify all gross outliers. We further introduce a way to speed up computation time without affecting the accuracy of model fitting.

II. Multiple Model Fitting

Our method is based on evolutionary dynamics for fitting multiple models to outlier corrupted and noisy data. In a first step, described in Section III, we follow the same procedure as in RANSAC and use a pre-defined number of randomly selected minimal sample sets to generate a set of model hypotheses, but in contrast afterwards switch to a conceptual representation to define pairwise similarities between the data points. In Section IV we then introduce the evolutionary dynamics to robustly fit multiple models to the data points by analyzing the obtained pairwise similarity matrix. Finally, in Section V we show promising performance in a line fitting experiment and demonstrate an application for fitting planes to a RGB-D stream.

III. Random Sampling and Conceptual Representation

The first step of our method is similar as in standard RANSAC procedures. We randomly generate M minimal sample sets (e. g. each set has three elements for a plane fitting task), sampling from the provided N data points. Then, we identify the so-called *consensus set* for each of the M models, i. e. we find all data points which have a distance to the model within a pre-defined threshold. Based on these consensus sets, we can build an $N \times M$ binary assignment matrix $\mathbf{A}$ with elements a_{ij}, where a_{ij} is 1 if data point i lies within the threshold range from the model j and 0 otherwise. We now consider each row $\mathbf{A}_{i.}$ as a novel, conceptual representation of the data points, each representing the assignments to the randomly hypothesized models. Since, as outlined before, data points belonging to the same model will have a similar conceptual representation, we can use this representation as basis for our model fitting procedure.

As next step, we build an $N \times N$ affinity matrix $\mathbf{M}$ by comparing the obtained conceptual representations $\mathbf{A}_{i.}$ using any available distance measure. For this reason, we define for each data point, its corresponding assignment set $\mathbf{S}_i$ by

$$\mathbf{S}_i = \left\{ s_i^1, s_i^2, \ldots s_i^{n_i} \right\} , \tag{1}$$

where the s_i represent the models that contain data point i within the defined threshold range and n_i is the number of elements in the set, i. e. $|\mathbf{S}_i| = n_i$. Based on these sets, we can apply any set-distance measure to define our affinity matrix $\mathbf{M}$. Following [10] we also use the Jaccard distance, which is an intersection-over-union measure defined as

$$M_{ij} = \frac{|\mathbf{S}_i \cup \mathbf{S}_j| - |\mathbf{S}_i \cap \mathbf{S}_j|}{|\mathbf{S}_i \cup \mathbf{S}_j|} , \tag{2}$$

where $|\mathbf{S}_i|$ is the number of elements in the set $\mathbf{S}_i$. In such a way, we finally provide the $N \times N$ affinity matrix $\mathbf{M}$, where data point pairs (i, j) belonging to the same model will have high affinity values M_{ij}, as input to the model fitting method as it is described in the next section.

IV. Evolutionary Dynamics for Model Fitting

The next step is to fit an undefined number of models to the data using the conceptual representation that was introduced in the previous section. Such a method has to be robust against noise and outliers and should not be adversely affected by overlapping models. Since data points belonging to the same model will have a similar conceptual representation, they will cluster together in the conceptual space. Thus, valid models can be identified by applying a proximity-based clustering approach on the obtained pairwise affinity matrix $\mathbf{M}$.

We analayze a wide-spread clustering criterion and aim at optimizing the intra-cluster coherency $f(\mathbf{x})$ for each fitted model defined as

$$f(\mathbf{x}) = \mathbf{x}^T \mathbf{M}\, \mathbf{x} \ , \tag{3}$$

where $\mathbf{x}$ is a model-specific indicator vector of size $N \times 1$, such that $x_i = 1$ if data point i is part of the fitted model and 0 otherwise. Our goal is to find the optimal indicator vector $\mathbf{x}^*$ for each model by optimizing

$$\mathbf{x}^* = \underset{\mathbf{x}}{\operatorname{argmax}}\, f(\mathbf{x}) \ . \tag{4}$$

The most common way to find a solution of such an optimization problem is to apply a spectral analysis on the matrix $\mathbf{M}$, as e. g. done in spectral clustering [11]. Nevertheless, in our model fitting scenario, it is important that this optimization step is robust, since we have to deal with a large number of outliers. For this reason, we propose to use dynamics from the field of *game theory* for this step that have shown to be more robust to outliers and to outperform spectral approaches [12], [13]. Using these dynamics we iteratively approach an *evolutionary stable strategy* (ESS) for each fitted model. Every ESS represents a set of data points having high intra-cluster coherency $\mathbf{x}^*$ according to the similarity of their conceptual representations, i. e. each ESS contains data points that fit to a common model. The different ESSs are found subsequently. Nevertheless, we introduce an approach that guarantees that each found ESS is a solution of the original formulation, thus the ordering in which the ESSs are found does not matter and even significantly overlapping models can be handled.

We first review the evolutionary process for maximizing the intra-cluster coherency in Section IV-A. Then in Section IV-B we outline our extensions for handling multiple models. Section IV-C describes how we can speed up our method while maintaining the overall model fitting performance.

A. Evolutionary dynamics

For fitting models, we propose to use evolutionary dynamics from the field of game theory, which is a tool for predicting how players behave in strategic situations (games). Each player has a set of available actions and the obtainable reward depends on the set of actions played by each player. In this paper we focus on a non-cooperative two player game, where players are in a competitive setting and the possible actions are adapting the probabilities of assigning the data points to the model. Final result of the non-cooperative game is an *evolutionary stable strategy (ESS)* which is the outcome of an evolutionary process that unfolds over time. Game theoretical approaches were recently gaining increased popularity in the field of computer vision, e. g. for matching image segments and points [14], for finding common spatial visual patterns in images [12] and for clustering [15], [13].

We first review *replicator dynamics* which is the currently most popular strategy to obtain an ESS in non-cooperative two player games. Replicator dynamics are a first order evolutionary dynamic from the field of game theory having several important properties like simple implementation and short computation time. The goal of the dynamics is to estimate an $N \times 1$ assignment vector $\mathbf{x}^*$ as the ESS (equilibrium) of the game, which implicitly maximizes the intra-cluster coherency $\mathbf{x}^*$ shown in Equation 3 and thus identifies a set of data points belong to the same (currently analyzed) model. Replicator dynamics are an iterative procedure defined as

$$x_i^{t+1} = x_i^t \ \frac{(\mathbf{M}\,\mathbf{x}^t)_i}{\mathbf{x}^{t^T} \mathbf{M}\,\mathbf{x}^t}, \tag{5}$$

where $\mathbf{x}^t$ is the assignment vector at time t. As a necessary additional constraint $\mathbf{x}$ has to lie on the simplex Δ defined as

$$\Delta = \left\{ \mathbf{x} \in \mathbb{R}^N : x_i \geq 0 \quad and \quad \mathbf{1}^T \mathbf{x} = 1 \right\}, \tag{6}$$

where $\mathbf{1}$ is an N-dimensional vector of ones, i. e. $\sum x_i = 1$. The dynamics start with a random initialization $\mathbf{\Pi}$ which also has to lie on the simplex. We always initialize the dynamics by a slightly perturbed version (added random noise) of the barycenter of the simplex. Starting from $\mathbf{\Pi}$, replicator dynamics find an optimal affinity vector $\mathbf{x}^* = \operatorname{argmax}_{\mathbf{x}} f(\mathbf{x})$ lying on the simplex which is a local (!) maxima of the coherency function $f(\mathbf{x})$. Finally, all entries of the solution $\mathbf{x}^*$ that have a value above zero, define the assignment of each data point to the current model. We define this set as the support $\sigma(\mathbf{x})$ by

$$\sigma(\mathbf{x}) = \{ i : x_i > 0 \} \ , \tag{7}$$

i. e. the support of the ESS directly defines the consensus set of the model. The simplex Δ is invariant under the replicator dynamics formulation, which means that every trajectory starting on the simplex (random initialization) will remain on the simplex. Furthermore, the coherency $f(\mathbf{x})$ is strictly increasing along any trajectory of the dynamics given in Equation 5. The final solution provided by the dynamics is the sought evolutionary stable strategy (ESS), which is a stricter formulation of the well-known Nash equilibrium from game theory. For more details and convergence proofs of these dynamics see e. g. [16].

The intuition behind such an approach is as follows: the hypotheses, that each data point belongs to the currently analyzed model, compete with each other. Each data point gains support from compatible points and competitive pressure from all other data points during the evolutionary process, where compatibility is defined according to the similarity of the conceptual representations. This competitive setting iteratively reduces the number of data points by driving inconsistent hypotheses to extinction, finding a model with high internal coherency, which is unaffected (!) by outliers. Once a model is found, we identify (similar as in RANSAC) the corresponding model parameters in a least-square manner using the identified inliers defined by the support $\sigma(\mathbf{x})$. Furthermore, the coherency as defined in Equation 3 can be directly used to quantify the model fitting quality, which enables to directly reject weak model hypotheses.

B. Multiple model fitting using evolutionary dynamics

It is important to note, that the dynamics as described before, only find a single evolutionary stable strategy (ESS) per run, i. e. only a single model is fitted. Since we aim at identifying all models, we have to apply the dynamics several times. The most simple approach would be to start the dynamics multiple times from random initial points, hoping to converge to different ESSs. However, obviously this is a quite inefficient way to explore the solution space. Another naive approach, as it was e. g. suggested in [16], is to use a peeling-off strategy, i. e. one can iteratively remove all inliers from the data after each step. Such an approach heavily resembles a sequential RANSAC concept, possessing the same drawbacks like incapability of handling overlapping models, since data points only contribute once in such a setting. Furthermore, such a sequential fitting procedure changes the optimization problem in every iteration, which might introduce ESSs that do not exist in the original problem formulation. Thus, we require a novel method, which maintains all inliers during multiple model fitting and directly returns all models having a coherency above a pre-defined threshold. Furthermore, we can guarantee that all provided solutions are ESSs of the original problem formulation.

We also use an iterative approach, but in contrast to the aforementioned approaches, we only adapt the values in the affinity matrix $\mathbf{M}$ after finding a valid model, *without removing* any already assigned data points. Adapting the affinity matrix $\mathbf{M}$ ensures that the optimization process subsequently addresses all valid models. In such a way, all data points possibly contribute to all extracted models, which enables to handle even severely overlapping models.

Let us assume that we already have found a solution $\mathbf{x}_a^*$ and its corresponding consensus set $\sigma\left(\mathbf{x}_{\mathbf{a}}^*\right)$, which defines the a-th model $\mathcal{M}_a$. We use the obtained solution $\mathbf{x}_a^*$ to downweight entries in the affinity matrix by

$$\mathbf{M}^{a+1} = \mathbf{M}^{a} - \lambda \left(\mathbf{x}_a^{*T} \mathbf{x}_a^*\right) , \qquad (8)$$

where λ is a parameter to fix how much emphasis should be put onto the update. We aim at preserving the internal energy (sum over all affinities) of the dynamics, by setting

$$\lambda = \frac{\sum\limits_{i=1}^{N} \sum\limits_{j=1}^{N} M_{ij}}{\sum\limits_{i=1}^{N} \sum\limits_{j=1}^{N} \left(\mathbf{x}_a^{*T} \mathbf{x}_a^*\right)} . \qquad (9)$$

Using the updated affinity matrix $\mathbf{M}^{a+1}$ we apply the evolutionary dynamics described in the previous section and identify our corresponding solution $\mathbf{x}_{a+1}^*$ for the next model.

It is important to note that, since we adapt the original affinity matrix $\mathbf{M}$ in this step, the obtained solution might not be a valid ESS of the original problem formulation. To overcome this issue, we inject our obtained solutions $\mathbf{x}_a^*$ into the simplex to define a new initialization vector $\overline{\mathbf{\Pi}}$ by

$$\overline{\Pi}_i = \begin{cases} 1/n_\sigma & \text{if } \; i \in \sigma\left(\mathbf{x}_a^*\right) \\ 0 & \text{else} \end{cases} , \qquad (10)$$

where n_σ is the number of elements in the support, i. e. $n_\sigma = |\sigma\left(\mathbf{x}_a^*\right)|$. We then apply the dynamics again, but this time using the original affinity matrix $\mathbf{M}$ and the updated initialization vector $\overline{\mathbf{\Pi}}$. The returned solution is then considered as the fitted model, and in such a way, we can *guarantee* that all obtained solutions are evolutionary stable strategies of the original problem formulation (since we are using the original affinity matrix). Thus, we iteratively obtain fitted models that all are solutions of our original problem formulation. This process is repeated until we obtain a model having a coherency below a fixed threshold δ. Model fitting results are not sensitive to δ, since invalid models have a significantly lower intra-cluster coherency. Please note, that after iteratively fitting the models, there might still exist data points that are not assigned to any of the models. All these points are directly identified as gross outliers, and we do not have to perform any error-prone outlier identification or cluster post-processing step, as it is e. g. required in [10], [9]. Algorithm 1 summarizes our algorithm in detail.

Algorithm 1: Multiple model fitting method

```
Input:  Data points $i$
Output: Model set $\mathcal{M} = \{\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_A\}$
Randomly generate model hypotheses and get
consensus set for each hypothesis
Build affinity matrix $\mathbf{M}$, analyzing conceptual
representation (Equation 2)
repeat
    Apply evolutionary dynamics (Section IV-A) to
    obtain solution $\mathbf{x}_a^*$
    if $a \neq 1$ then
        Inject solution into original problem formulation
        and apply dynamics to obtain final inlier set
        $\sigma\left(\mathbf{x}_a^*\right)$ (Equation 10)
    Do Least Square Fitting on inlier set to get model
    parameters for $\mathcal{M}_a$
    Update affinity matrix (Equation 8)
until obtained model has coherency below $\delta$;
return All models $\mathcal{M}_a$ and corresponding inliers $\sigma\left(\mathbf{x}_a^*\right)$
```

C. Efficiency improvements

If we directly apply the dynamics as presented before we have a complexity quadratic in the number of data points N. Nevertheless, we can significantly speed up the dynamics by exploiting the properties of the task that we aim to solve, which is quite similar to the idea proposed in PROSAC [4].

The core insight is, that for obtaining local maxima of the coherency function, not all other data points have to participate in the game, i. e. have to contribute to the dynamics. This has a straight forward intuition that data points of a valid model group quite closely together using our conceptual representation as similarity measure. Furthermore, data points having a low pairwise affinity do not crucially influence the evolutionary dynamics shown in Equation 5. Thus, the number of data points considered could be limited, allowing to significantly reduce computation time without affecting the model fitting quality.

Therefore, we make our input affinity matrix $\mathbf{M}$ sparse and only consider the K nearest neighbors of each data point,

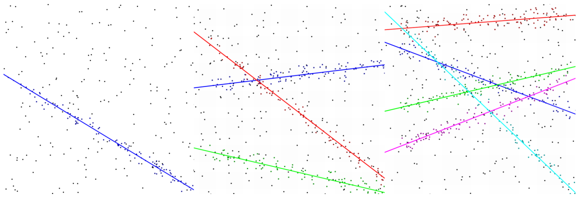

Fig. 2: Synthetically generated line fitting examples. Number of outliers (0 to 250), σ of Gaussian noise (0.0025 to 0.025) and number of lines (1 to 5) are adapted.

which reduces the complexity of the dynamics from N^2 to $N\,K$, where $K \ll N$. Please note, that the nearest neighbors can be easily identified during building the affinity matrix. In fact as it is shown in the experimental section, K can be fixed to only a fraction of N (e. g. 5%) resulting in the same model fitting quality. In such a way, by fixing K to a small number, we can significantly speed up our model fitting method.

V. Experiments

We evaluate our proposed method on synthetic and real world data. In Section V-A we describe an experiment for line fitting on synthetically generated 2D point sets and evaluate our contribution concerning the nearest neighbor based sparsification of the affinity matrix, demonstrating that the same model fitting quality is achievable at reduced computation time. Finally, in Section V-B we demonstrate the applicability of the proposed method for real-time plane fitting on video streams obtained by the Microsoft Kinect sensor.

A. Fitting multiple lines

This experiment analyzes randomly generated synthetic 2D line fitting examples. Some typical examples are visualized in Figure 2. Each test case includes a number $1 \leq K \leq 5$ of randomly generated lines each consisting of 100 inliers points which are contaminated with Gaussian noise of varying σ from 0.0025 to 0.05. Additionally 0 to 300 gross outlier points are randomly inserted in our considered image range of $[0\ 1] \times [0\ 1]$. While by adapting the variance σ and the number of outliers L we can test the robustness of the analyzed methods concerning noise and outliers, increasing the number of lines K aims at evaluating how well the methods cope with overlapping models.

We test the performance of our method by analyzing 10 repetitions for each parameter setting. Our method consistently identifies the correct number of lines yielding a fitting accuracy compareable to RANSAC in all cases, until the number of outliers is beyond 240 or the noise level is higher than 0.04. Starting from these values performance breaks down, because the algorithm fails to fit the right number of lines.

Furthermore, we demonstrate the applicability of our proposed concept for reducing computation time as described in Section IV. We use the same experimental setup as described before, this time analyzing the fitting error for different percentage of data points considered in each dynamic evolution process. Experiments showed that results stay the same down to considering only **5%** of the data points as nearest neighbors, which leads to a speed-up of approximately 40 in our implementation. Further reducing K below 5% degrades performance a lot, since then the data point affinities are not connected anymore, which prevents correct model fitting.

B. Plane fitting on video streams

As main experiment, we applied our proposed method for fitting planes to 3D data points sampled from a video stream obtained by the Microsoft Kinect depth sensor using videos from the *RGB-D Dataset and Benchmark* [17], which is a recently released benchmark for the evaluation of visual SLAM systems. The data set provides color and depth images of a Microsoft Kinect sensor recorded at full frame rate (30 Hz) and sensor resolution (640x480), which makes it suited for our plane fitting task.

For evaluation we equidistantly sample points within the range of 0 to 5 m from the camera center in each frame. We afterwards apply our multiple model fitting procedure on the corresponding 3D world coordinates of the sampled points. We fixed the inlier threshold to 5 cm, the number of hypothesized models to 1000, the number of nearest neighbors K to 5%, the minimum size of a consensus set to 40 and the coherency threshold δ again to 0.25.

We show the corresponding videos for plane fitting on these streams in the additional material. Some exemplary results selected from different sequences of the benchmark are shown in Figure 3. The individual planes are highlighted by different markers and are colored based on the plane normal, so that parallel planes get assigned the same color. Please note, that each frame is processed in a completely independent manner, i. e. no type of tracking is considered. As can be seen our method accurately finds the dominant planes in the sequences. Our current implementation in Matlab runs in approximately 1 frame per second, which would presumably enable real-time plane fitting on Microsoft Kinect depth sensors using an optimized implementation.

VI. Conclusion

In this paper we proposed a novel method for fitting multiple models to data points corrupted with outliers and noise. Our method fits model hypotheses by analyzing the conceptual representation of data points using evolutionary dynamics from game theory. Robust fitting results are obtained while we are able to guarantee that all obtained models are solutions of a wide-spread intra cluster coherency maximization problem, which is unaffected by outliers. In such a way, we are able to handle even severely overlapping models. We further described a way to significantly reduce the computation time by considering only nearest neighbors in the evolutionary process, nevertheless yielding the same model fitting accuracy. Experiments for fitting lines to synthetically generated 2D point sets demonstrated promising results, and we furthermore showed accurate results for fitting multiple planes to video streams obtained by the Microsoft Kinect depth sensor.

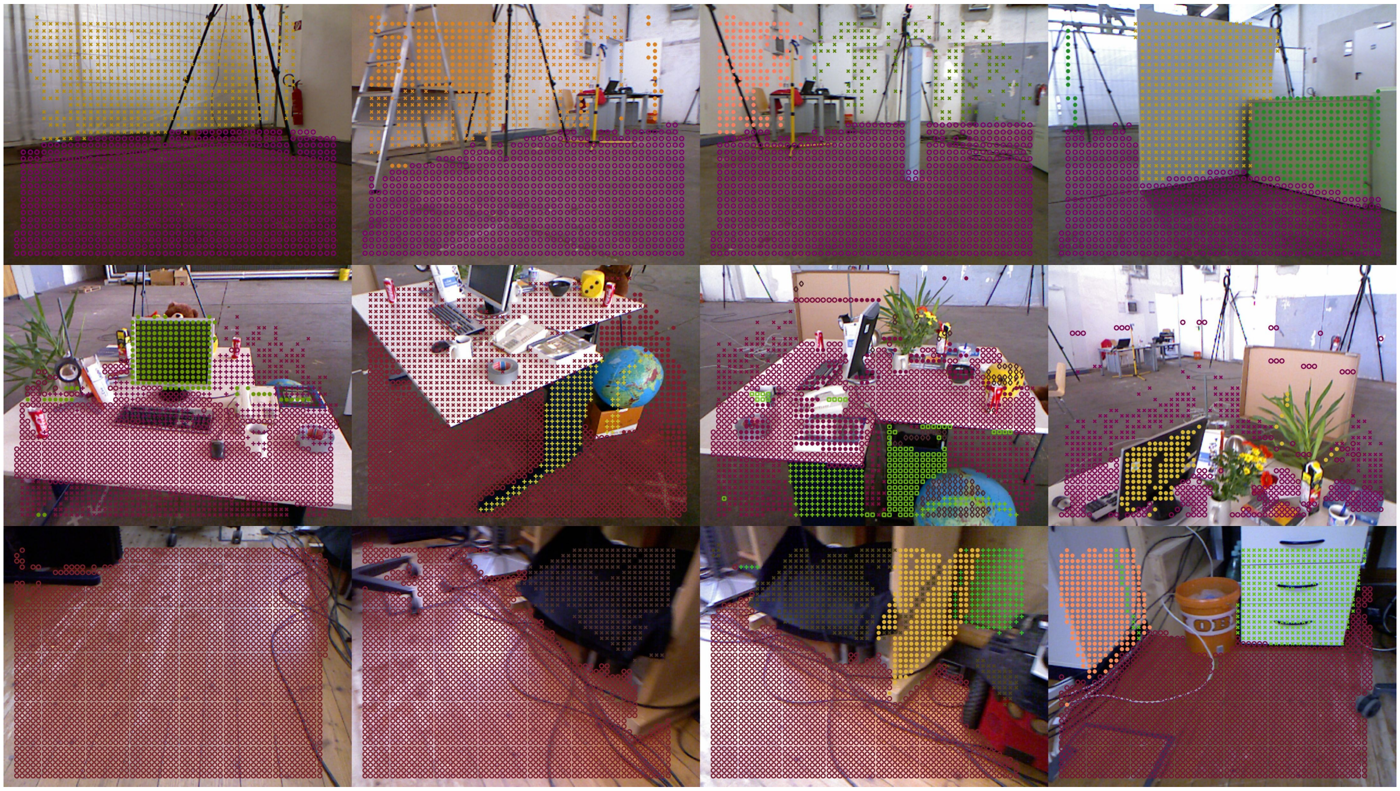

Fig. 3: Fitting planes to a video stream obtained by the Microsoft Kinect depth sensor. The identified planes are highlighted by different markers and are colored based on the plane normal, so that parallel planes get assigned the same color (best viewed in color).

Acknowledgment

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 601139 CultAR (Culturally Enhanced Augmented Realities).

References

[1] M. A. Fischler and R. C. Bolles, “Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography,” *Communications of the ACM*, vol. 24, pp. 381–395, 1981.

[2] R. Raguram and J.-M. Frahm, “Recon: Scale-adaptive robust estimation via residual consensus,” in *Proceedings of International Conference on Computer Vision (ICCV)*, 2011.

[3] M. Zuliani, C. S. Kenney, and B. S. Manjunath, “The multiransac algorithm and its application to detect planar homographies,” in *Proceedings International Conference on Image Processing (ICIP)*, 2005.

[4] O. Chum and J. Matas, “Matching with prosac - progressive sample consensus,” in *Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[5] L. Xu and E. Oja, “Randomized hough transform (RHT): Basic mechanisms, algorithms, and computational complexities,” *CVGIP: Image Understanding*, vol. 57, no. 2, pp. 131 – 154, 1993.

[6] J. Yu, T. J. Chin, and D. Suter, “A global optimization approach to robust multi-model fitting,” in *Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[7] A. Delong, A. Osokin, H. Isack, and Y. Boykov., “Fast approximate energy minimization with label costs,” *International Journal of Computer Vision (IJCV)*, 2011.

[8] W. Zhang and J. Kosecka, “Nonparametric estimation of multiple structures with outliers,” in *Workshop on Dynamic Vision at European Conference on Computer Vision*, 2006.

[9] T.-J. Chin, H. Wang, and D. Suter, “Robust fitting of multiple structures: The statistical learning approach,” in *Proceedings of International Conference on Computer Vision (ICCV)*, 2009.

[10] R. Toldo and A. Fusiello, “Robust multiple structures estimation with j-linkage,” in *Proceedings of European Conference on Computer Vision (ECCV)*, 2008.

[11] A. Y. Ng, M. I. Jordan, and Y. Weiss, “On spectral clustering: Analysis and an algorithm,” in *Advances in Neural Information Processing Systems*, 2001, pp. 849–856.

[12] H. Liu and S. Yan, “Common visual pattern discovery via spatially coherent correspondences,” in *Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[13] ——, “Robust graph mode seeking by graph shift,” in *Proceedings of International Conference on Machine Learning (ICML)*, 2010.

[14] A. Albarelli, S. Bulo, A. Torsello, and M. Pelillo, “Matching as non-cooperative game,” in *Proceedings of International Conference on Computer Vision (ICCV)*, 2009.

[15] S. R. Bulo, A. Torsello, and M. Pelillo, “A gametheoretic approach to partial clique enumeration,” *Image and Vision Computing*, vol. 27, pp. 911–922, 2009.

[16] M. Pelillo, “Matching free trees with replicator equations,” in *Advances in Neural Information Processing Systems (NIPS)*, 2001.

[17] J. Sturm, S. Magnenat, N. Engelhard, F. Pomerleau, F. Colas, W. Burgard, D. Cremers, and R. Siegwart, “Towards a benchmark for RGB-D SLAM evaluation,” in *Proceedings of the RGB-D Workshop on Advanced Reasoning with Depth Cameras at Robotics: Science and Systems Conference (RSS)*, 2011.