Energy-based Geometric Multi-Model Fitting

Hossam Isack   Yuri Boykov

Computer Science Department

University of Western Ontario, Canada

`habdelka@csd.uwo.ca,`   `yuri@csd.uwo.ca`

March 30, 2010

(IJCV submission, June 2010)

Abstract

Geometric model fitting is a typical chicken-&-egg problem: data points should be clustered based on geometric proximity to models whose unknown parameters must be estimated at the same time. Most existing methods, including generalizations of RANSAC, greedily search for models with most inliers (within a threshold) ignoring overall classification of points.

We formulate geometric multi-model fitting as an optimal labeling problem with a global energy function balancing geometric errors and *regularity* of inlier clusters. Regularization based on spatial coherence (on some near-neighbor graph) and/or label costs is NP hard. Standard combinatorial algorithms with guaranteed approximation bounds (e.g. α-expansion) can minimize such regularization energies over a finite set of labels, but they are not directly applicable to a continuum of labels, e.g. $\mathcal{R}^2$ in line fitting. Our proposed approach (PEARL) combines model sampling from data points as in RANSAC with iterative re-estimation of inliers and models parameters based on a global regularization functional. This technique efficiently explores the continuum of labels in the context of energy minimization. In practice, PEARL converges to a good quality local minima of the energy automatically selecting a small number of models that best explain the whole data set. Our tests demonstrate that our energy-based approach significantly improves the current state of the art in geometric model fitting currently dominated by various greedy generalizations of RANSAC.

Keywords: geometric models, a-expansion, graph cuts, sampling

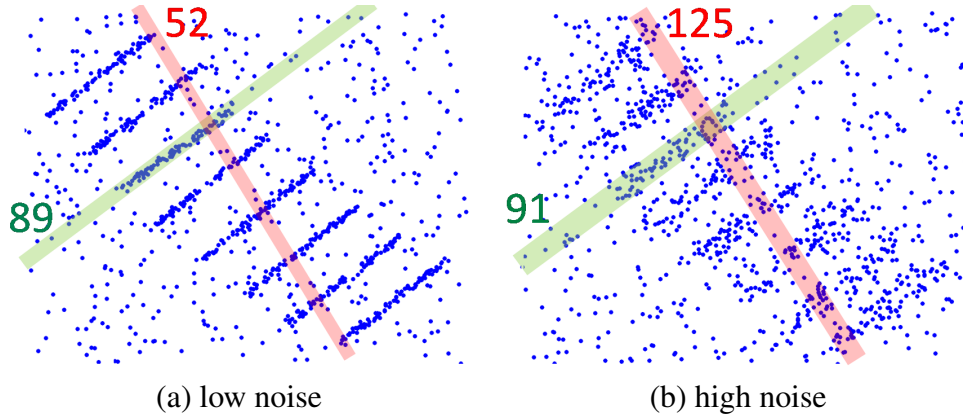(a) low noise   (b) high noise

Figure 1: Blue dots are data points supporting 8 lines. In multi-model cases, detecting models by maximizing the number of inliers may work for low levels of noise (a). Higher noise levels require larger thresholds to detect inliers (b), but then, some random model (red) may have far more inliers than the true model (green). The integers show the number of model inliers for selected thresholds.

1 Introduction

We study a general case of geometric multi-model fitting problem where data is a mixture of outliers with points supporting unspecified number of models of some known type[1]. The majority of existing algorithms treat inlier classification and parameter estimation as isolated subproblems. Typically, each model is selected greedily by maximizing inliers within some fixed threshold. Popularized by RANSAC [10], this approach works well when data supports a single model, but we argue that it is fundamentally flawed in multi-model cases, see Fig.1.

RANSAC [10] is a well-known robust method for dealing with large number of outliers when data supports only one model. The main idea is to generate a number of model proposals by randomly sampling data points and then select one model with the largest set of inliers (a.k.a. consensus set) with respect to some fixed threshold. Many publications [24, 27, 30] proposed various generalizations of RANSAC for multi-model fitting. For example, [24, 27] run RANSAC sequentially. Each iteration of these methods selects one randomly sampled model maximizing either the number of inliers or some similar threshold-based measure. Thereby identified model's inliers are removed from the set of data points before

---

[1]For simplicity, we assume (parametric) models of same type. This is not essential.
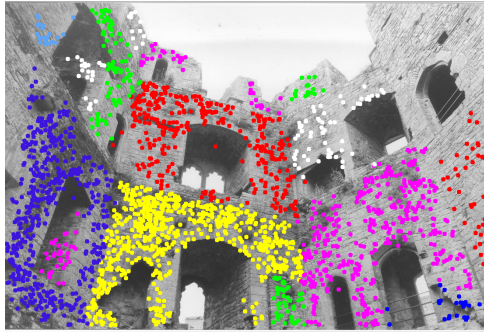
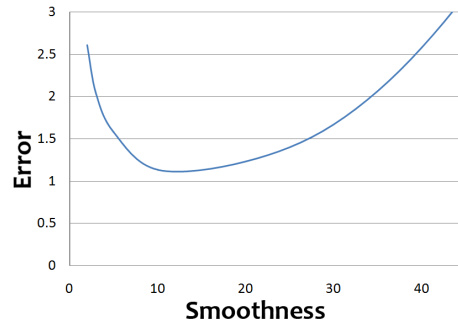(a) fitting homographies (stereo)   (b) estimation accuracy vs. λ

Figure 2: Motivating spatial regularization in geometric model fitting. In many vision problems combining geometric errors and spatial coherence terms in energy (3) can be justified *generatively* because clusters of inliers are generated by regular objects (a). More over, spatial regularization can also be justified *discriminatively*. Plot (b) shows an average deviation from the ground truth for optimal models obtained in 100 randomly generated line-fitting tests as in Fig.4d. Each point in this plot corresponds to some fixed smoothness parameter λ in (3). Clearly, spatial coherence term significantly reduces estimation errors for some $\lambda > 0$.

the next iteration looks for the next model. Other methods rely on different forms of greedy clustering, e.g. J-linkage [21], explicitly or implicitly maximizing the number of inliers within given threshold. One can also apply Hough transform to formulate multi-model fitting as a clustering problem in the space of model parameters and use *mean-shift* [6] to identify the modes in this Hough space. It is easy to see that this approach also greedily maximizes the number of inliers.

We argue that greedy maximization of inliers is a flawed criteria for detecting a model if data supports multiple models. Figure 1 shows simple examples illustrating the main problem: if the level of noise is increased, any random model can easily have a larger number of inliers than the true models. This also explains our experimental results in Section 3 (Figs.7-9) demonstrating that existing methods work reliably only for examples with very low levels of noise and clutter.

1.1   Towards energy optimization

This paper argues that geometric multi-model fitting is better formulated as an optimization problem with a global energy functional describing the quality of

the overall solution. An energy function sets some specific "goodness" criterion for different solutions and the sought optima can be seen as "objectively" the best solution with respect to this criterion. There are many problems in computer vision (e.g. segmentation, optical flows, stereo) routinely solved as optimization problems. Yet, we know only a few examples in vision [24, 2, 16] where some specific geometric multi-model fitting problems were approached using an energy-based formulation. Perhaps, limitations of these methods (see Sec.1.2) restrict their general use and explain why many researchers in the community still use greedy heuristics for geometric multi-model fitting problems. Our goal is a general energy-based framework for geometric multi-model fitting problems with efficient algorithms and wide applicability in computer vision.

There are several reasons why standard energy-based methods for mixture models, such as EM or K-means, are not very common in geometric multi-model fitting in vision. In general, these methods may not be robust to outliers/noise. More importantly, they depend on a priori knowledge of the number of models. For example, EM framework in MLESAC [23] is fixed to 2 models (inliers/outliers), and [11] estimate known number of motions in cases with relatively low noise[2]. Soft assignment of inliers by EM algorithm could be questioned as a valid approach to geometric problems in vision where models must have spatially non-overlapping support, see Fig.2a. K-means is also known to have a bias towards equally dividing the points among the models.

In order to motivate our general approach, we first demonstrate some energy-based interpretation of the basic RANSAC algorithm [10]. This interpretation is limited to a simple case when data supports only one model (e.g. one line). The main goal of RANSAC is to find parameters L of the model with the largest number of inliers within some threshold T. This can be represented as minimization of energy

$$E(L) = \sum_{p} ||p - L||$$

where

$$||p - L|| = \begin{cases} 0 & \text{if } dist(p, L) < T \\ 1 & \text{otherwise} \end{cases}$$

and $dist(p, L)$ is Euclidean distance between data point p and the nearest point on model L. In this paper $||p - L||$ will generally denote an arbitrary error measure for point p and geometric model L. RANSAC's energy $E(L)$ counts inliers for

---

[2]From personal communications with A. Gruber.

L using 0-1 measure $||p - L||$ above. Typically, RANSAC also includes an additional step refining model parameters L by minimizing the sum of squared errors for inliers. Thus, a more principled optimization-based formulation of RANSAC leads to MSAC energy [23] using truncated Euclidean errors

$$||p - L|| = \begin{cases} dist^2(p, L) & \text{if } dist^2(p, L) < T \\ T & \text{otherwise.} \end{cases}$$

Note that RANSAC or MSAC optimize $E(L)$ only over model parameters L and inliers are identified *implicitly* from threshold T in the corresponding error measures $||p - L||$.

Now assume that data supports multiple models. If the number of models is known (say K) it could be possible to formulate geometric model-fitting as optimization of energy $E(L_1, L_2, ...L_K)$ over K model parameters. As in the earlier example with a single model, this approach needs some implicit assignment of inliers to models. In multi-model case, however, this could be non-trivial. As shown in Fig.3b, simple thresholding may assign a point to several models. Interestingly, the EM framework for mixture models [3, 23, 11] corresponds to energy $E(L_1, L_2, ...L_K)$. EM uses implicit "soft" classification of inliers computed in an intermediate optimization step. Even though the standard version of EM algorithm needs the number of models to be known, there are many generalizations of EM that could be worth studying in the context of geometric applications in vision. However, we prefer to focus on a fairly different energy formulation based on explicit "hard" classification of inliers. As shown in Fig.2a, in many problems in computer vision geometric models have non-overlapping spatial support, which better corresponds to hard assignment of inliers.

We formulate geometric multi-model fitting as an optimal labeling problem. Consider the general case when the data supports some unknown number of models. In principle, in this case each data point p can have a separate model L_p. Model fitting could be formulated as minimization of energy $E(\mathbf{L})$ over labeling $\mathbf{L} = \{L_p | p \in P\}$ of points in data set P. Since labeling $\mathbf{L}$ explicitly assigns models to data points, inliers support $\{p | L_p = L\}$ for any specific model L does not have to be implicitly deduced from some threshold.

If the only goal is to minimize the model fitting errors, as in the example with one model, then

$$E(\mathbf{L}) = \sum_p ||p - L_p||. \tag{1}$$

where $||p - L||$ could be an arbitrary error measure. Obviously, this functional would not work well as the globally optimal solution will independently fit some

(a) one line   thresholding   (b) three lines   thresholding
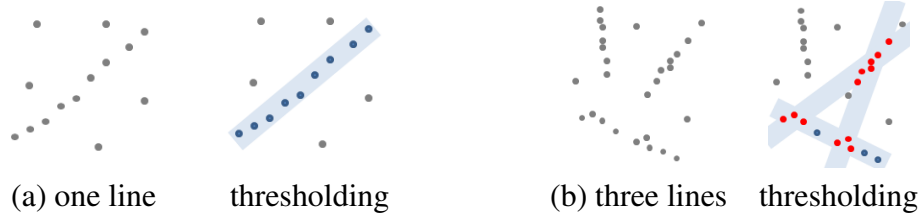
Figure 3: Examples of inlier classification from thresholding. If data is known to support one model (a) then thresholding identifies inliers (blue) for any model without ambiguity. In case of 3 models (b), simple thresholding may not disambiguate inliers (red) between the models.

model L_p to each point p. This corresponds to overfitting: every point is assigned some perfectly fit model and there are no outliers. It is clear that model fitting errors (1) must be combined with some energy term regularizing the labeling. One special "outlier" label could be added as well.

One form of regularization for (1) could be to fix the number of allowed distinct models/labels. Then, energy (1) corresponds to the standard K-means algorithm. This approach, however, does not work if the exact number of models is not known a priori. Recently, Li [16] proposed a soft form of regularization for the number of models by combining geometric errors with the label count penalty

$$E(\mathbf{L}) = \sum_p ||p - L_p|| + \beta \cdot |\mathcal{L}_{\mathbf{L}}| \qquad (2)$$

where $\mathcal{L}_{\mathbf{L}}$ is the set of distinct models (labels) assigned to points by labeling $\mathbf{L}$. Ten year earlier Torr [24] suggested even more general form of such regularization where each distinct model (label) gets a penalty defined by the model's complexity instead of some fixed constant β. This approach allows to fit models of different types. In general, geometric model-fitting using energies like (2) is a very interesting idea, but specific algorithms for minimizing such energies proposed in [24] and [16] are fairly limited, see Sec.1.2. We also argue that spatial regularity of inliers is required in many typical vision problems, see Fig.2a.

This paper proposes two specific general forms of regularization in the context of geometric model fitting. In particular, we consider spatial regularization (3)

$$E(\mathbf{L}) = \sum_p ||p - L_p|| + \lambda \cdot \sum_{(p,q) \in \mathcal{N}} w_{pq} \cdot \delta(L_p \neq L_q),$$

where $\mathcal{N}$ is some neighborhood (e.g. edges on some near-neighbor graph), and a more general combination of spatial smoothness with label counts (5)

$$E(\mathbf{L}) = \sum_p ||p - L_p|| + \lambda \cdot \sum_{(p,q) \in \mathcal{N}} w_{pq} \cdot \delta(L_p \neq L_q) + \beta \cdot |\mathcal{L}_{\mathbf{L}}|.$$

While spatial regularization is very common in vision in general, it is not common in geometric model fitting. In part, this could be explained by the fact that spatial coherence is hard to justify generatively in applications where data points are i.i.d. samples. But in computer vision, see Figure 2a, one can defend generative models of spatial regularity. Figure 2b also suggests that spatial regularization may work discriminatively even for i.i.d. data[3].

Summary of contributions   This work demonstrates significance of efficient combinatorial optimization methods for a wide range of geometric applications in computer vision. Surprisingly, such methods are overlooked in geometric model fitting, even though they are very common in segmentation, dense stereo, optical flows, and other problems. We believe that we contribute a new approach and significant algorithmic ideas specific to general geometric multi-model fitting. We see our two main contributions as follows:

- We propose a general practical energy-based framework for geometric model fitting particularly suitable for a wide range of applications in vision. To the best of our knowledge, energies (3) and (5) were not used for general geometric model fitting problems in the past. We demonstrate conceptual advantages and significant practical improvements over the state-of-the-art methods on a large number of generic applications in vision (line/plane fitting, homography estimation, rigid motion detection). In particular, we argue against typical greedy heuristics currently dominating in geometric model fitting and hope that our work would encourage a wider use of energy optimization methods well known in other areas of computer vision.

- Energies like (3) and (5) can be addressed with existing powerful combinatorial optimization techniques with guaranteed optimality bounds (e.g. α-expansion [5]) only in cases of finite set of labels. This limits their use

---

[3]One can not expect spatial regularization to work well for i.i.d. data, in general. Line fitting examples in Section 2 are a special case where it does work for i.i.d. points. We use these line fitting examples only to illustrate the basic operations of our algorithm. The primary target of our model fitting approach are applications in vision (Sec.3) where spatial coherence is well-founded.

for geometric model fitting where the typical space of model parameters is a continuum, e.g. $\mathcal{R}^2$ in line fitting. We propose a practical method (PEARL) for efficiently exploring the continuum of labels (model parameters) in the context of energy-based geometric model fitting.

Our approach alleviates dependence of many previous geometric model fitting methods on thresholding. The proposed methods for optimizing energies (3) and (5) work quite differently from greedy selection of models by the largest number of inliers. Our approach is robust to high levels of noise and clutter. It automatically computes on optimal set of labels/models with a good fit to data points.

In order to apply standard discrete optimization algorithms to energies (3) and (5), we generate a large number of proposed labels (models) by sampling data points as in RANSAC. The goal of this step is to prune the search space (the continuum) of model parameters. As in RANSAC, the number of sampled models should be sufficiently large to guarantee with some level of confidence that at least one sample was generated from inliers for each true model. Such finite pool of labels is likely to contain good model candidates. However, in contrast to RANSAC-style methods we rely on optimization of a global energy functional to select some small subset of models from this large (but finite) pool of proposals. Exploration of the continuum of labels (model parameters) is further significantly improved by iterating inlier segmentation for a finite set of labels and re-estimation of these labels (model parameters) from their inliers. Both steps minimize the same energy $E(\mathbf{L})$ and correspond to a coordinate descent converging to a local minimum of the energy. Such iterative refinement of model parameters and inlier classification allows to generate better solutions from a smaller set of initial samples even in single model fitting, see Fig.6.

1.2   Related work

Our work proposes, justifies, and validates a wide class of regularization energies and a powerful iterative optimization technique as a general framework for geometric multi-model fitting particularly suitable in vision. Other geometric model fitting works have used separate elements of our approach such as RANSAC-style random sampling [24, 16] or EM-style iterations [2], but none have combined them in a single optimization framework. We also use a more general form of regularization (5) than any earlier geometric fitting methods. Our experiments show that our general energy-based approach beats many state-of-the-art algorithms in this area. In other settings (segmentation, stereo) some elements of our framework

have been used in various application-specific ways [29, 2, 20, 28].

Probably the earliest effort to formulate some general energy-based framework for geometric model fitting in vision is due to Torr [24] who carefully justifies a general version of energy like (2) from an *information criterion*. The specific optimization algorithm in [24] was a greedy heuristic very similar to *sequential RANSAC* also described in [27] a few years later. Figures 1 and 9 show that such heuristics can generate solutions arbitrarily worse than good minima of the energy.

Birchfield & Tomasi [2] estimate affine geometric models in a way fairly specific to dense narrow base-line stereo. They combine photoconsistency of pixels with spatial regularization (as in (3)) on a grid. The method is initialized from a disparity map generated by another stereo algorithm. The most noticeable overlap of our approach and [2] is iterative use of α-expansion and model re-estimation steps. After [2] and [20] such EM-style optimization became fairly common for different problems in vision. In contrast to [2], however, our framework is suitable for a significantly more general set of geometric problems. For example, instead of photoconsistency we optimize geometric errors and combine them with more general forms of regularization, e.g. (5). Our method is more concerned with fitting to sparse data. Finally, we do not need to run other algorithms to initialize. Our experiments in Sec.3.1 show noticeable improvement of accuracy on examples from [2].

Zabih & Kolmogorov [28] also use iterative optimization as in [2, 20] specifically in the context of image segmentation. They use spatial regularization (as in (3)) to cluster image pixels into spatially coherent segments with automatically estimated color models. The color models are initialized in an application specific way. In contrast, we work with a very different problem of geometric model fitting studying more general regularization functional (5). In fact, our recent work [8] with additional coauthors demonstrates that energy (5) may significantly improve image segmentation results. Minimum description length (MDL) interpretation of (5) is well known in segmentation literature for some time [15, 29].

The paper by Li [16] is probably the most closely related prior work. Similar to [24], it formulates general geometric model fitting functional (2) and studies it in the context of rigid motion estimation, which we also consider as one of the applications in Sec.3.3. Instead of the greedy approach of [24], [16] uses LP relaxation of (2). This could be slow. To speed up the method, [16] uses several application specific heuristics to significantly prune the set of proposed models. More importantly, [16] does not guarantee any optimality of the discrete solution obtained after rounding and the quality of such optimization could be an issue. These problems do not allow [16] to use EM-style iterative optimization

that, in our experience, can significantly improve model fitting results. A better optimization of energy (2) with some optimality guarantees is discussed in [8].

In this paper, we argue that (5) is generally a better energy for geometric model fitting problems in vision. We found that per-label regularization term proposed in [24, 16] is a practically useful addition to standard spatial regularization (3). Fig.10 shows one illustrative example where penalty for using each distinct label encourages the merging of isolated clusters supporting nearly identical models. Similarly, the results on real motion detection sequences in Figure 19 fail to merge spatially isolated background clusters into one motion if label counts are not a part of the energy. To optimize the third term in energy (5) one can use a simple and fast merging step in combination with standard α-expansion optimizing the first two terms in the energy. This merging heuristic is discussed at the end of Sec.2. Alternatively, [8] provides an extension of α-expansion algorithm that automatically accounts for the third term in (5) incorporating it into each expansion step as a high-order clique. The technical details of such extension is a subject of [8]. The main focus of this paper is to demonstrate that a general algorithmically solid optimization approach to geometric model fitting with either (3) or (5) is a significantly better alternative to greedy generalizations of RANSAC-style thresholding currently dominant in geometric problems in vision.

The structure of our paper   Section 2 presents our general method for fitting multiple models to sparse data points. For simplicity, most of the details are explained in the context of energy (3). Energy (5) is introduced in the end of the section. Section 3 provides evaluation on real data in narrow-base stereo, wide-base stereo/reconstruction, and rigid motion estimation.

2   Our approach (PEARL)

This section described our algorithm for geometric model fitting in detail. For simplicity, the main ideas are illustrated in the context of synthetic multi-line fitting examples. Section 3 validates our approach for estimating affine transformations, homographies, and rigid motion models in the context of computer vision.

We use regularization labeling framework to assign models to data points. Regularization energy can combine geometric fit errors with spatial smoothness term (3) and a label count penalty (5). As long as the number of labels is finite ($\leq$ 10000 or so), such approach can be handled by graph-based optimization methods,

e.g. a-expansion [5, 8]. In our case the labels are models described by n real-valued parameters. Therefore, we should find a practical way to restrict the huge search space of labels $\mathcal{R}^n$. The first step is to *propose* a finite set of plausible models (labels). In the next step each label is *expanded* to estimate its spatial support, or to classify inliers. Once the inliers are fixed, the models (labels) can be *re-estimated* by minimizing the geometric errors - the first term in our energy functionals. As both *expand* and *re-estimate* are guaranteed to decrease the energy, one can converge to a local minima by iterating over these two steps. One can also iterate *propose* steps either by further sampling the data points, or by generating some new proposed labels (models) from the currently supported models, e.g. by merging them. Below we provide more details about our model-fitting algorithm mostly in the context of energy (3). Energy (5) is introduced in the end of the section.

2.1   *Propose* **initial labels** $\mathcal{L}_0$

First, our method uses random sampling of data points to *propose* an initial finite set of models $\mathcal{L}_0 \subset \mathcal{R}^n$, where n is the number of parameters describing each model ($n = 2$ for lines and $n = 6$ for affine models). The idea of generating models by sampling the data points is borrowed from RANSAC [10]. The required number of initial models $|\mathcal{L}_0|$ is one of the parameters of RANSAC-based methods. It depends on the number of data points, the number of outliers, the number of estimated models, the minimum number of points requited to estimate each model, and on desired level of confidence. The exact analysis for the case of estimating a single model is given in [10]. Its adaptation to multi-model case is in [27, 30]. The number of initial models $|\mathcal{L}_0|$ for PEARL is analyzed in [14]. Our experiments suggest that in practice PEARL needs far fewer samples than the theoretical estimate due to converging iterations that significantly improve probability of an accurate model reconstruction from rough initial guesses.

2.2   Energy formulation

Once initial finite set of proposed models $\mathcal{L}_0 \subset \mathcal{R}^n$ is known (see Fig.4(a)), we can *expand* the models to estimate their spatial support. We use MRF-based regularization framework and a-expansion optimization [5] to assign models to data points. The set of current models in $\mathcal{L}_0$ is interpreted as a set of current labels. Assume that P is a set of data points and that $L_p \subset \mathcal{R}^n$ is a label (model) assigned to a given data point $p \in P$. Then, PEARL method estimates models

(a) initial 25 model proposals   (b) models & inliers (iteration 1)

(c) models & inliers (iteration 2)   (d) at convergence (iteration 5)
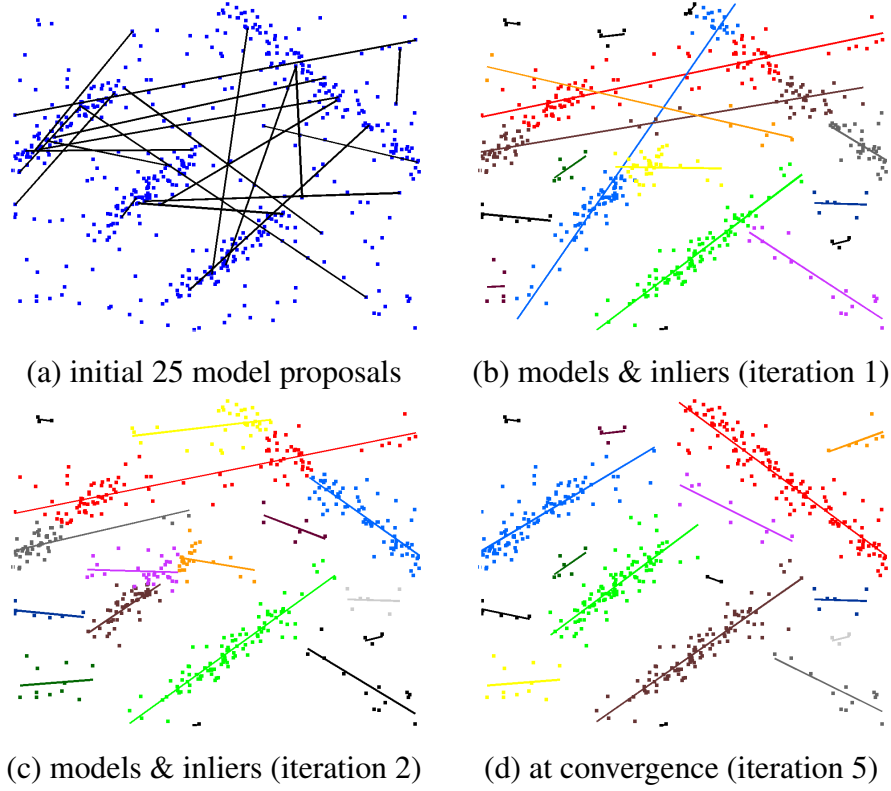
Figure 4: Illustration of PEARL's iterations. (a) proposals generated by random sampling, (b-d) re-estimation of models and their inliers after several iterations of *expand* and *re-estimate* steps for energy (3). Note that the algorithm can converge to good model estimates from a small number of rough guesses.

and their spatial support (inliers) by optimizing the following energy of labeling $\mathbf{L} = \{L_p | p \in P\}$

$$E(\mathbf{L}) = \sum_p ||p - L_p|| + \lambda \cdot \sum_{(p,q) \in \mathcal{N}} w_{pq} \cdot \delta(L_p \neq L_q). \qquad (3)$$

The first term $||p - L||$ in (3) measures geometric error between point p and model L. For example, the line fitting examples in this section use "perpendicular distance" between 2D point $p = (x, y)$ and line $L = (a, b)$

$$||p - L|| = \left(\frac{|y - ax - b|}{\sqrt{a^2 + 1}} \right)^2$$

which is the distance from p to the nearest point on line L. Robust (truncated) measures are also possible. Term $||p-L||$ corresponds to the log-likelihood $\ln \Pr(p|L)$ when energy (3) is interpreted as an MRF-based posterior energy. Thus, the use of quadratic distance for $||p-L||$ is equivalent to assuming Gaussian distribution for errors. Clearly, optimal labeling L for (3) depends on specific choice of geometric measure $||p-L||$.

The second term of energy (3) is a smoothness prior. It assumes some specific neighborhood system $\mathcal{N}$ for the data points. For example, the neighborhood system could be based on a triangulation of points, see Fig.5. In this paper we use the Potts model, e.g. [5], where $\delta(\cdot)$ is 1 if the specified condition inside parenthesis holds, and 0 otherwise. Weights w_{pq} set discontinuity penalties for each pair of "neighboring" data points. For example, the synthetic line fitting examples in this section use weights w_{pq} inversely proportional to the distance between points p and q because closer points are a priori more likely to fit the same model

$$w_{pq} = \exp -\frac{||p-q||^2}{\sigma^2}.$$

In all of our experiments σ was constant and it was chosen heuristically to be 5. Examples in Section 3 use constant weights $w_{pq} = 1$. Besides Potts (piecewise constant) prior, one can also consider piece-wise smooth priors. Such priors would allow small variation of model parameters between data points.

2.3   *Expand* and *re-estimate* labels

Energy (3) can be minimized using α-expansion algorithm [5] for labels $\alpha \in \mathcal{L}_0$. In this case, it is possible to interpret α-expansions as a competition among model-labels for spatial support; models with the best-fit to data points find the largest number of spatially coherent "inliers", while most of the "erroneous" models get no inliers.

Once inliers are computed, model labels in $\mathcal{L}_0 \subset \mathcal{R}^n$ with non-empty set of inliers can be re-estimated as follows. Note that the first term of energy (3) can be represented as

$$\sum_p ||p - L_p|| = \sum_{L \in \mathcal{L}_0} \sum_{p \in P(L)} ||p - L||$$

where $P(L) = \{p \in P|L_p = L\}$ denotes a set of inliers for label L. Clearly, we can minimize this expression by re-estimating parameters of each model $L \in \mathcal{L}_0$

$$\hat{L} = arg \min_l \sum_{p \in P(L)} ||p - l||. \tag{4}$$

(a) "Lines" data, Fig.8 a   (b) "Clorox" data, Fig.12 b
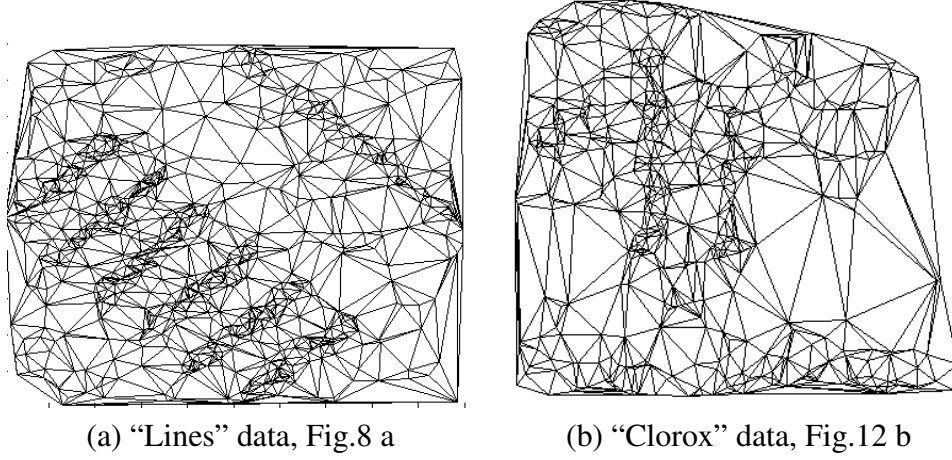
Figure 5: We use Delaunay triangulations of data points. Other techniques can be used, particularly for higher dimensional data.

We replace each label L with non-empty support $P(L)$ by label $\hat{L} \in \mathcal{R}^n$ which has a better fit to points in $P(L)$. Finally, after discarding all labels with no inliers, we obtain a new set of labels $\mathcal{L}_1$. Note that this operation does not affect the second (smoothness) term in (3) unless two labels L, L' become equal after re-estimation $\hat{L} = \hat{L}'$ (in this case, the smoothness energy also decreases). Clearly, the described operation of changing the set of labels

$$\mathcal{L}_0 \to \mathcal{L}_1$$

can only decrease the energy (3).

There are many known methods for optimizing the sum of geometric errors $||p - L||$ in (4). Optimization method may depend on specific choice of measure $||p - L||$. For example, to minimize the sum of squares of vertical-shift errors in our lines-fitting examples we use standard SVD-based methods. A large number of other examples of geometric or algebraic error measures $||p - L||$ and different methods for optimizing them are widely discussed in computer vision literature, e.g. see [13]. Our approach can incorporate many of these error functions $||p-L||$.

Figure 4 (b) visualizes clusters of inliers and re-estimated models $\mathcal{L}_1$ obtained in two separate steps described above: *expand* (inlier classification) and *reestimate* (model parameters). In some cases it could be useful to iterate the *propose* step as well. For example, new labels can be generated by merging or splitting clusters

of inliers. One interesting example of "merging" is described in the context of example in Figure 10 in the end of this section.

2.4   Algorithm and its properties

Both *expansion* (inlier classification) and *reestimation* steps decrease energy (3). Thus, we can iterate over these steps until convergence, see Fig.4 (b-d). We can stop the iterations when a new round of a-expansion does not change inliers. As soon as the spatial support of the current models (labels) stops changing, re-estimation of the models (4) can not improve geometric error term. It is clear that this iterative algorithm converges to some local minima. PEARL algorithm (Propose Expand and Re-estimate Labels) is summarized here:

1) **Propose**:
   — at initialization, randomly sample data to obtain $\mathcal{L}_0$, set i=0
   —* (optional for $i > 0$) sample more or merge/split current models in $\mathcal{L}_i$

2) **Expand**:
   — run α-expansion [5][4] for energy (3) and $\alpha \in \mathcal{L}_i$
   — if the energy does not decrease, stop

3) **Re-estimate Labels**:
   — solve (4) and obtain a new set of labels $\mathcal{L}_{i+1}$
   — set i = (i+1), go to step 2 (or optional to *).

Figure 6 shows an empirical plot showing how accuracy of the estimated model depends on the number of random samples. The errors are averaged over many randomly generated line-fitting examples. In these tests the data supported only one line model so that we can compare our approach with standard RANSAC. Our point is that iterations of *expand* and *re-estimate* steps can converge to a good estimate of the model from a rough initial guess. In contrast, RANSAC relies on a larger number of samples to find an accurate estimate.

Figures 7-9 compare PEARL to existing geometric multi-model fitting methods [30, 21] and mean-shift [6] which were discussed in the introduction. The synthetic multi-line examples were generated using different levels of Gaussian noise and different number of uniformly distributed outliers. The previous methods were tuned to get the best results for each specific level of noise and clutter.

---

[4]Using extension of α-expansion [8], this step also works for energy (5) introduced later.
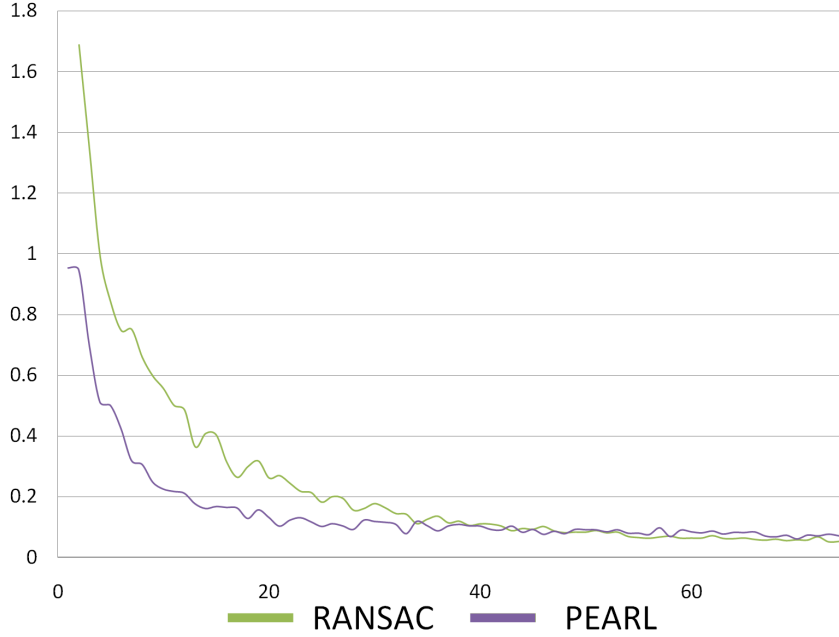
Figure 6: PEARL vs RANSAC for data sets with one line model. X-axis is the number of randomly sampled models. Y-axis shows estimation errors for the line parameters, which are averaged over 400 randomly generated tests. As illustrated in Figure 4, PEARL can converge to a good model estimate from a rough initial guess by iterating *expand* and *re-estimate* steps. In contrast, RANSAC relies on a larger number of samples to find an accurate estimate.

To demonstrate robustness of PEARL, we only changed one parameter σ in the geometric error measure $||p - L|| = -\ln G_\sigma(p - L)$ where $G_\sigma(\cdot)$ is a Gaussian distribution function and $p - L$ is the distance from point p to line L. Note that PEARL obtains very similar results when parameter σ is automatically estimated as described in the context of example in Figure 11.

For PEARL and multiRANSAC the data points were uniformly sampled while for J-linkage and mean-shift we used distance-based sampling which helps J-linkage and mean-shift algorithms. Sampling closer points increases the probability that the sampled model is closer to one of the peaks in the Hough transform.

Mean-shift and J-linkage have no constraints on the number of models they generate. Figures 7-9 show their strongest 7 models. MultiRANSAC had to
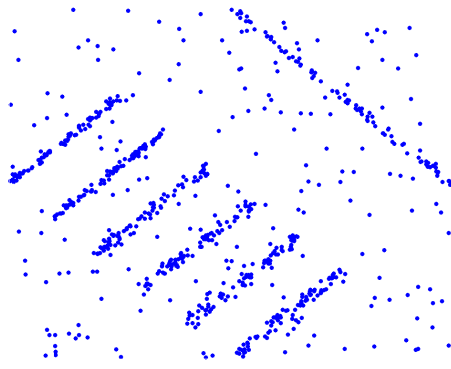
(a) The data points (200 outliers)
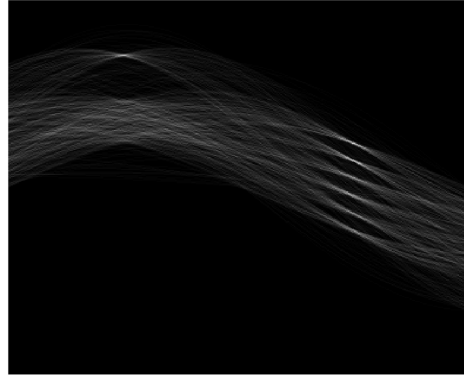
(b) The Hough transform of the data
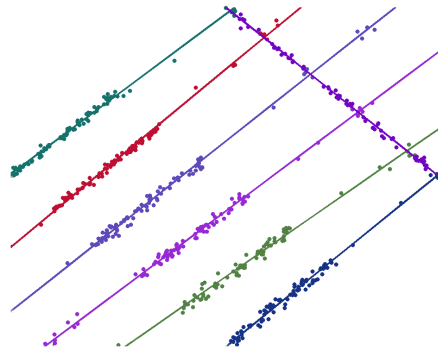
(c) MultiRANSAC result
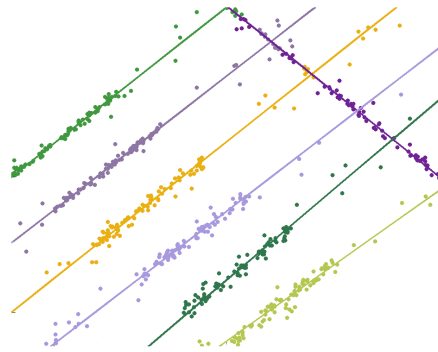
(d) mean-shift result
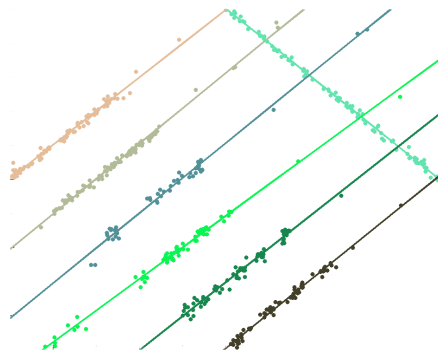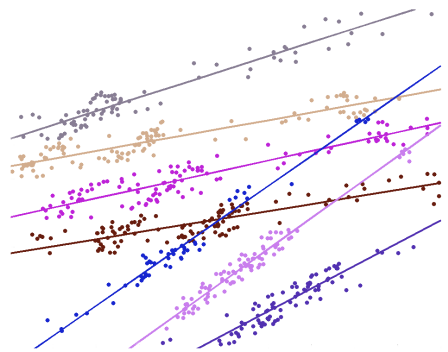
(e) J-Linkage result
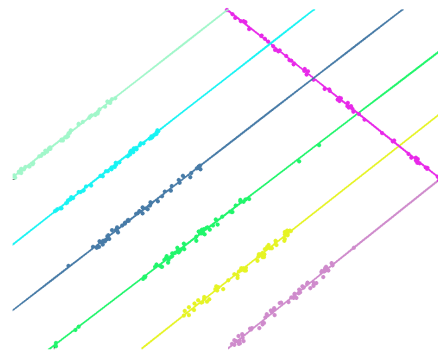
(f) PEARL result

Figure 7: Comparing the results for fitting lines to noisy data points. The data points were perturbed with a low level of Gaussian noise ($\sigma = 0.01$) and 200 outliers were added. Outliers represent 25% of the data.
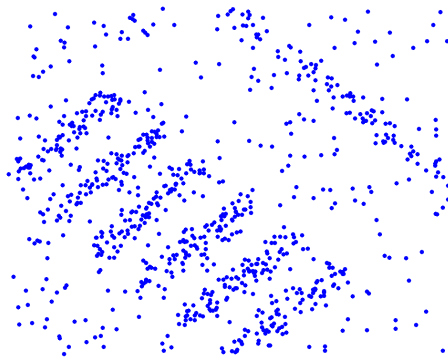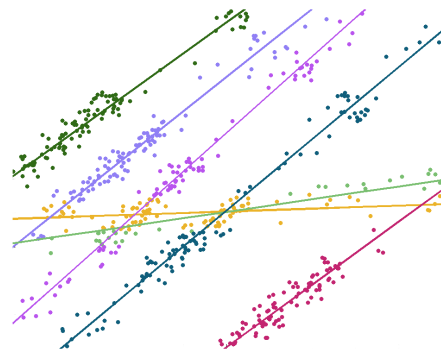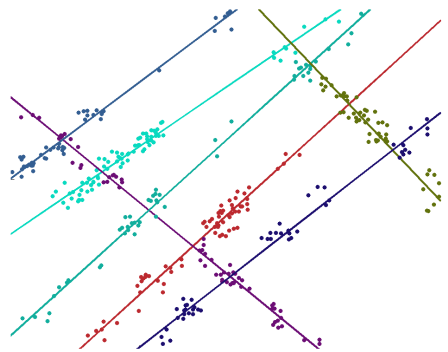
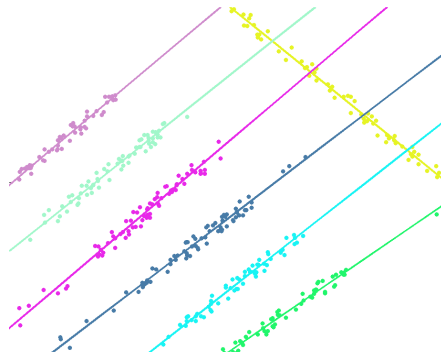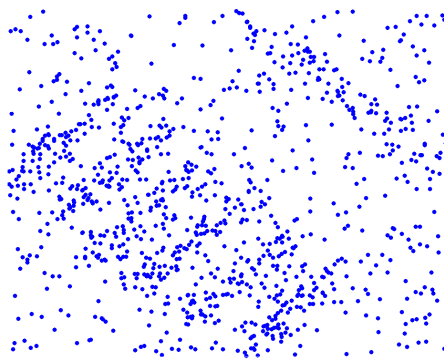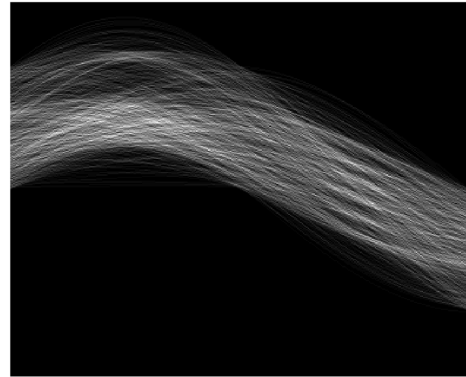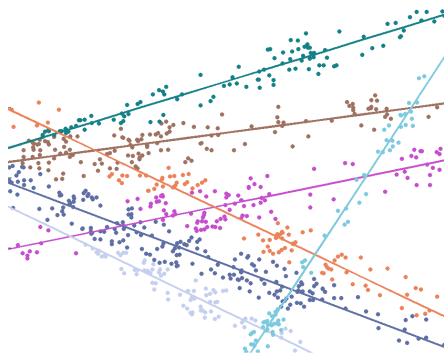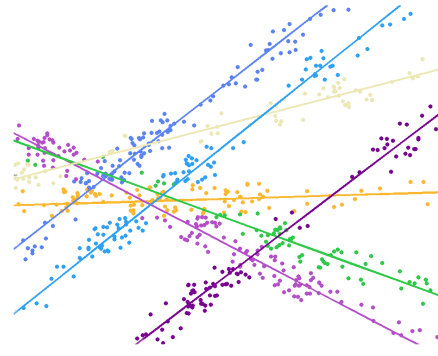(a) The data points (300 outliers)
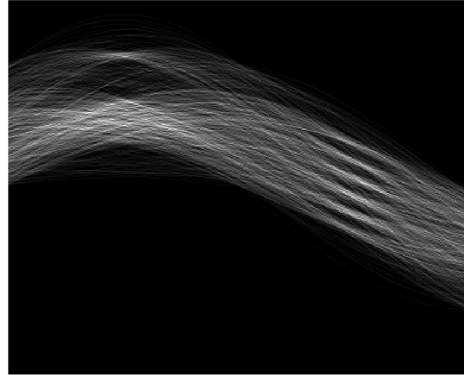
(b) The Hough transform of the data

(c) MultiRANSAC result

(d) mean-shift result

(e) J-Linkage result

(f) PEARL result

Figure 8: Comparing the results for fitting lines to noisy data points. The data points were perturbed with a low level of Gaussian noise ($\sigma = 0.02$) and 300 outliers were added. Outliers represent 25% of the data.
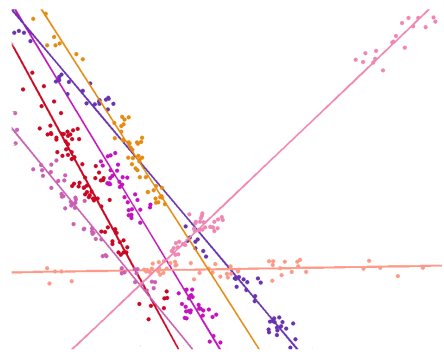
(a) The data points (600 outliers)
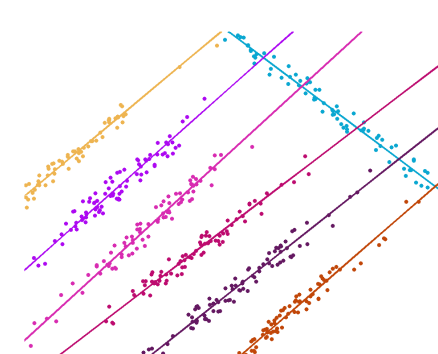
(b) The Hough transform of the data

(c) MultiRANSAC result

(d) mean-shift result

(e) J-Linkage result

(f) PEARL result

Figure 9: Comparing the results for fitting lines to noisy data points. The data points were perturbed with a low level of Gaussian noise ($\sigma = 0.025$) and 600 outliers were added. Outliers represent 45% of the data.

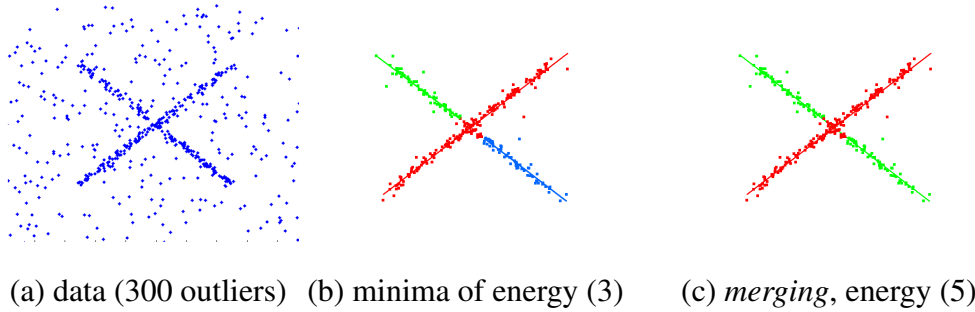(a) data (300 outliers) (b) minima of energy (3)   (c) *merging*, energy (5)

Figure 10: Intersecting lines example. Optimization of energy (3) may leave spatially isolated groups of inliers assigned to 2 models even if their parameters are infinitesimally close (b). Per-label costs in energy (5) justify our postprocessing merging step (c). This example may also suggest EM-style soft assignments of labels to points near the intersection. However, in vision (Sec.3) we get occlusions (not intersections), which better motivate our "hard" assignments of labels.

be given the exact number of models. Compared to mean-shift and J-linkage, PEARL finds a very small number of models giving with the optimal fit to the data. But, in addition to correctly identified true models, it can "hallucinate" a few models among outliers (as in Fig.4d). Such weak models can be automatically filtered out by setting a very conservative limit on the minimum number of inliers.

Figures 7-9 confirm that detecting models by maximizing the number of inliers is a flawed criteria in multi-model cases. As illustrated in Figure 1, random models may have more inliers than the true ones. This explains why the previous methods work relatively well only for the low noise example in Figure 7.

As we discussed in the introduction, coherence between inliers is often a good assumption particularly for problems in vision. Figure 10 shows one typical example where this assumption could be violated. Clearly, one of the intersecting lines cannot be assigned spatially connected group of inliers. More over, optimal solution for energy (3) can not merge two models with very similar parameters if their inliers are spatially separated (Fig.10b). However, a simple postprocessing step can merge separated groups of inliers with similar models (Fig.10c) if the "average" optimal model increases the sum of geometric errors by no more than some predefined threshold β. In fact, this merging operation can be seen as an optimization step if the energy function gets an additional term penalizing the

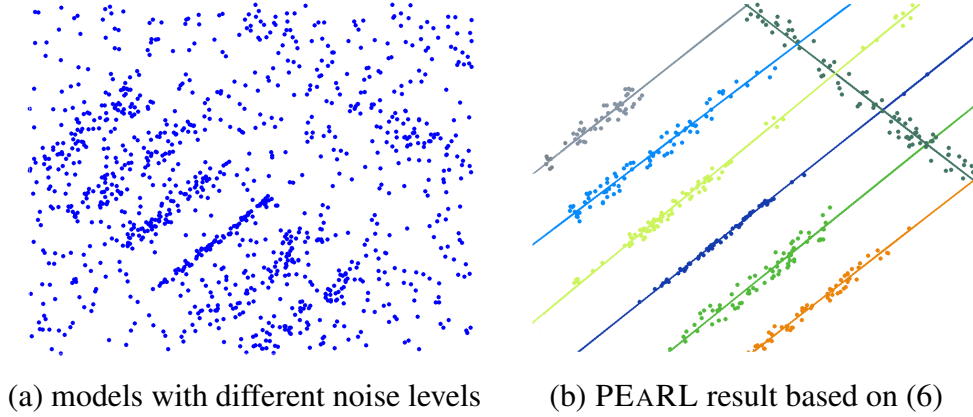(a) models with different noise levels   (b) PEARL result based on (6)

Figure 11: Fitting lines with different noise levels. The inliers in (a) were generated with different levels of Gaussian noise. 40% of the data are outliers. In (b) PEARL estimated labels combining geometric model parameters with unknown noise variances using error measure (6). Previous multi-model fitting methods use fixed thresholds to identify inliers, which would not work in this case.

number of models/labels with non empty support $|\mathcal{L}_\mathbf{L}|$

$$E(\mathbf{L}) = \sum_p ||p - L_p|| + \lambda \cdot \sum_{(p,q)\in\mathcal{N}} w_{pq} \cdot \delta(L_p \neq L_q) + \beta \cdot |\mathcal{L}_\mathbf{L}|. \qquad (5)$$

Instead of the proposed merging heuristic, energy (5) can be also minimized using an extension of α-expansion algorithm [8]. Note that ideas in [24] allow to generalize the label costs in energy (5) in order to work with models of different complexities. This extension of our optimization framework is straightforward and left as a simple exercise for the reader.

Figure 11 demonstrates another interesting feature of our optimization approach. Unlike many previous multi-model fitting methods [27, 30, 27] using fixed thresholds, PEARL can identify multiple models with different levels of noise. For example, this can be achieved as follows. Assuming that geometric errors for inliers correspond to Gaussian noise, one can set geometric error penalty $||p - L_p||$ according to the negative logarithm of the normal distribution function

$$||p - L_p|| = -\ln G_\sigma(p - L_p)$$

where $p - L_p$ is the distance from p to the assigned model L_p. Here one assumes some known σ parameter corresponding to the distribution's variance. If models

come with unknown different levels of noise, one can estimate extended labels $\tilde{L}_p = \{L_p, \sigma_p\}$ combining geometric model parameters L_p with the corresponding unknown noise-level σ_p. In this case one can use error measure

$$||p - \tilde{L}_p|| = -\ln G_{\sigma_p}(p - L_p). \tag{6}$$

Figure 11 shows that this approach can correctly estimate both geometric and statistical noise-level parameters for each model.

3 Experimental results

This section validates our model fitting technique (PEARL) on multi-view reconstruction data sets supporting multiple models. Our experiments used affine models (Sec.3.1) and homographies (Sec.3.2). Data points were obtained by matching SIFT[17] features on rectified image pairs in narrow-based stereo, and on uncalibrated wide-base pairs.

3.1 Estimating multiple affine models

In this section we apply PEARL to estimate affine transformation in the context of rectified narrow-base stereo. We use SIFT [17] features as points of interest, since they are scale and rotation invariant. They are also partially invariant to illumination and 3D camera view point changes. Matches between pairs of points in two images are found using exhaustive search along the corresponding scan line. In principle, it is possible to replace exhaustive search with "smarter" methods as in [1, 19].

We will use the notation (x_l, y_l), (x_r, y_r) to describe the coordinates of the image feature on the left image p_l and right image p_r. The symbol p denotes a pair of matching points (x_l, y_l, x_r, y_r).

A planar homography has only three degrees of freedom, in case of rectified images. As the epipole $\mathbf{e} = [1 \ 0 \ 0]^T$ in that case is at infinity. In turn the fundamental matrix could be formulated as $F = [\mathbf{e}]_x$, we use the notation $[\mathbf{e}]_x$ to describe the skew-symmetric matrix of the vector e. Since a planar homography must satisfy the following constrain $H^T F + F^T H = 0$ [9].That will enforce the planar homography for rectified stereo pair to be a special case of an affine trans-

formation, with only degrees of freedom as follows

$$A = \begin{pmatrix} a & b & c \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

we will refer to that special affine transformation as affine transformation for simplicity.

That type of transformation requires only three matching pairs to be uniquely identified. We first generate the finite set of initial model proposals A_0 by random sampling three matching pairs and computing their corresponding model parameters. One way to measure geometric error between a matching pair of points p and a given model A is

$$||p - A|| = (ax_l + by_l + c - x_r)^2 \tag{7}$$

which is a horizontal shift (along the epipolar line) between p_r and Ap_l.

The next step is to triangulate the features of one of the images (e.g. the right image). Then we apply PEARL to energy

$$E(\mathbf{A}) = \sum_p ||p - A_p|| + \lambda \sum_{(p,q) \in \mathcal{N}} \delta(A_p \neq A_q)$$

where $\mathbf{A} = \{A_p | p \in P\}$ is an assignment of affine models to data points p.

Figures 12 (a-c) compare affine model fitting results regenerated by BT [2] and two results generated by PEARL for different geometric error measures $||p - A||$. BT [2] use dense segmentation of pixels based on photoconsistency. This measure does not work well in texturelss regions and they have to rely on intensity edges (static cues) to detect the boundaries between regions supporting different models. In contrast, PEARL labels a sparse set of distinct features based on geometric (or algebraic) errors and spatial proximity.

Our results in Figures 12 (b,c) demonstrate that specific choice of geometric measure $||p - A||$ affects the results. For example, horizontal shift error in equation (7) worked well for all vertical planes in the scene, but it split the ground plane into two separate planes, see Fig.12(b). The problem with horizontal error (7) is illustrated in Figure 13. Basically, it can significantly overestimate the distance between a triangulated point and non-vertical planes in the 3D scene. Note that the ground plane in "Clorox" data set is the only non-vertical plane in the scene.
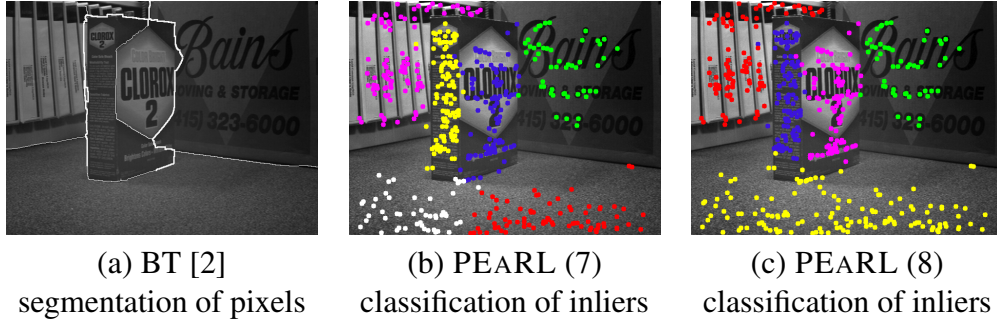
(a) BT [2]	(b) PEARL (7)	(c) PEARL (8)
segmentation of pixels	classification of inliers	classification of inliers

Figure 12: Results for "Clorox" stereo pair [2]. (a) Dense pixel segmentation by BT [2] uses photoconsistency. (b,c) Sparse inlier classification by PEARL using geometric fit measures (7,8).

To improve the results for this plane, we computed PEARL result (Fig.12(c)) for a different geometric error measure

$$||p - A|| = \frac{(ax_l + by_l + c - x_r)^2}{1 + w \cdot b^2} \tag{8}$$

that weights down the horizontal shift error for non-vertical planes ($b > 0$). This error is derived from the fact that allowing some vertical shift for both corresponding points p_r and p_l to a different epipolar line adds an additional degree of freedom in computing the shortest distance to the perfect alignment with affine model A. This significantly reduces the error for non-vertical planes while not changing the error for vertical ones ($b = 0$).

In order to provide some quantitative comparison between the affine models generated by BT [2] and PEARL, we used a "ground truth" image (Fig.14(a)) where we manually extracted the lines corresponding to intersecting planes. Assuming that two intersecting planes π_1 and π_2 are represented by the affine models A^{π_1} and A^{π_2}, the homogenous vector representing the line of intersection is defined as the first row of the matrix $(A^{\pi_1} - A^{\pi_2})$. Therefore, such lines can be computed from the models estimated by either BT or PEARL. Table 1 compares geometrically the accuracy of these lines.

3.2   Estimating multiple homographies

In this section we apply PEARL to estimate multiple homographies for uncalibrated wide-base stereo image pairs. We use SIFT [17] features as points of
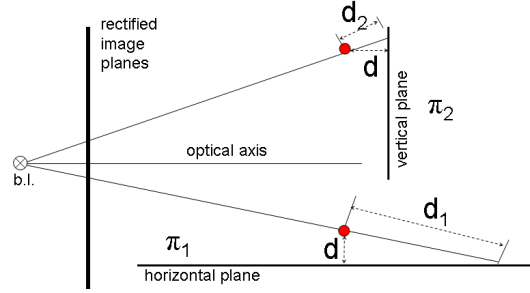
Figure 13: Interpretation of horizontal shift error (7). The 3D scene is displayed from a side view-point orthogonal to images' optical axis. Assuming that two red points are at the same distance from the image plane, one can easily check that error (7) is proportional to d_1 and d_2 shown above. Therefore, even if two points are at the same distance d from planes π_1 and π_2, the horizontal shift error is much larger for the horizontal plane.

Line	BT [2]	PEARL (7)	PEARL (8)
L_1	34.55	8.80	9.46
L_2	16.83	4.82	3.66
L_3	5.56	13.27	6.26
L_4	5.99	4.46	12.05
Total	62.94	31.37	31.45

Table 1: Geometric errors for lines in Fig.14 where affine models intersect. Errors are computed as the sum of distances between the ground truth line segment corners and the computed lines.
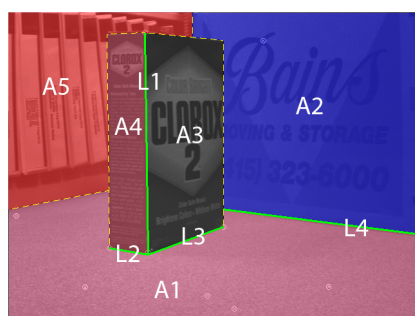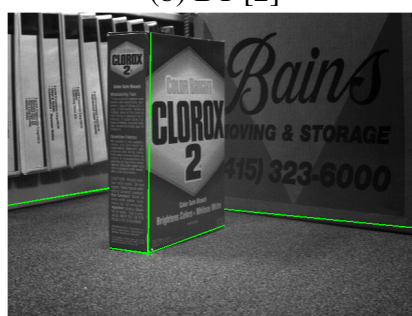
(a) "Ground truth"
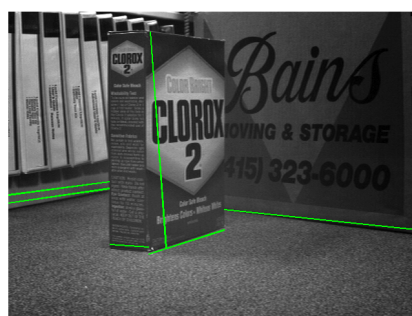
(b) BT [2]

(c) PEARL (7)

(d) PEARL (8)

Figure 14: Comparison of results by BT [2] and PEARL. Lines are computed for all pairs of intersecting planes (affine models).
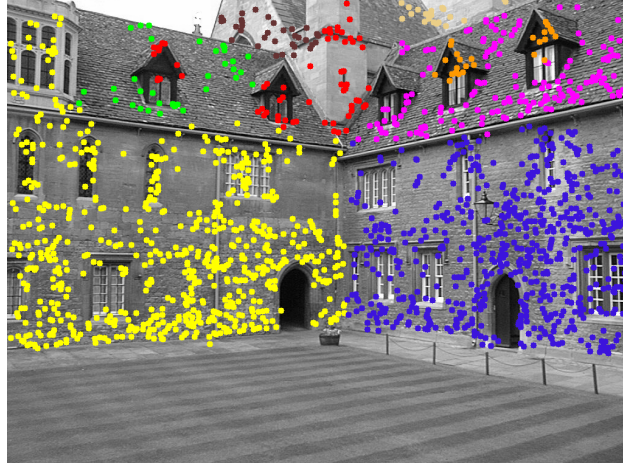
Figure 15: Stereo images from VGG (Oxford) Merton College I

interest. The set of all matching pairs P is found using exhaustive search.

One way to measure geometric error between a matching pair of points p and a given model H is the Symmetric Transfer Error (STE) [13]. We generate our finite set of initial model proposals H_0 by randomly sampling four matching pairs. Then compute the model parameters as descried in [13] by minimizing the non-linear STE error using Levenberg-Marquard. The initial solution for the non-linear minimization is found using the Direct Linear Transform method.

The next step is to triangulate the features of one of the images (e.g the right image).Then we apply PEARL to energy

$$E(\mathbf{H}) = \sum_{p} ||p - H_p|| + \lambda \sum_{(p,q)\in\mathcal{N}} \delta(H_p \neq H_q)$$

where $\mathbf{H} = \{H|p \in P\}$ is an assignment of models to data points p.

In the example shown in Fig.15 PEARL was able to identify 8 planes with large and small spatial support. Unlike multiRANSAC [30] PEARL does not require the prior knowledge of the number of planes. Also PEARL produces spatially coherent set of inliers for each plane compared to multiRANSAC. In [30] multiRANSAC required 11604 iterations for the same image set, in each of these iterations 4 random samples were drawn sequentially, meaning that the multiRANSAC required a total of 46416 random samples in order to find just 4 planes. On the other hand PEARL using only 900 initial random samples was able to identify 8 planes. PEARL required only three iterations to converge.
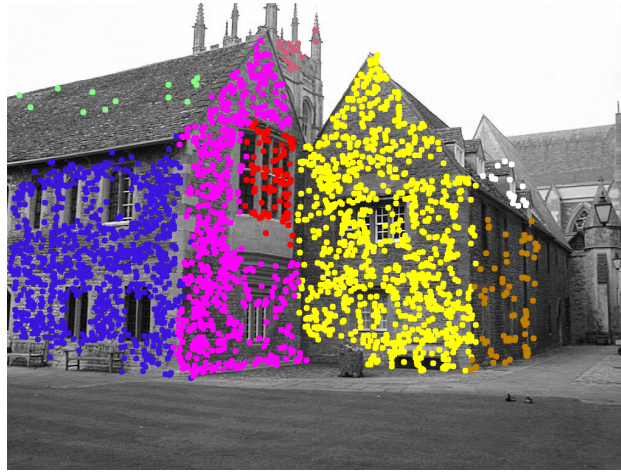
Figure 16: Stereo images from VGG (Oxford) Merton College III.

In the example shown in Fig.16 PEARL was able to identify 8 planes. Only 3000 initial labels were used, PEARL converged after four iterations. Notice how planes with varying number of inliers could easily be identified using PEARL. The roof of the building on the left (shown in light green) was supported only by 13 inliers, while the two neighboring large planes (shown in blue and pink) were supported by 786 and 581 inliers respectively.

Figure 17 shows Raglan Castle Tower. Using 6000 initial labels and only four iterations to convergence. PEARL was able to identify 13 planes. The use of relatively large number of initial labels allowed PEARL to identify very small planes, e.g the top planes of the third floor. Later on we found out from another picture for Raglan Castle Tower on flicker, that each of the second and third floors are shifted to the inside by one step from the floor underneath it.

3.3   Motion Segmentation

In this section we aim to solve the multibody motion segmentation problem using multiple-views. This problem is also referred to in literature as the multibody structure from motion problem [4, 22, 7]. The goal of this problem is to cluster the scene trackable features among distinct motions, then to estimate the motions' parameters and finally to recover the 3D structure of the points. In our case we are only interested in estimating the multiple motions and clustering the image
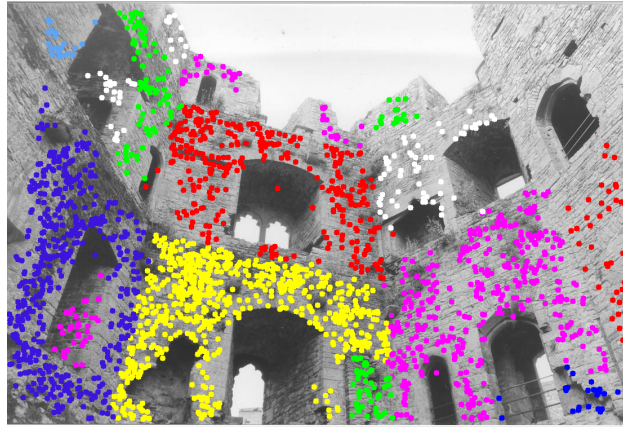
(a) Multiple planes detected by PEARL

(b) A different point of view (by anonymous Flicker photographer)

Figure 17: In this Raglan Castle example from VGG (Oxford) we used the same color more than once to represent different planes (a). Only spatially connected planes are shown in different colors. An image of the same scene from a different view (b) confirms that each floor of the building corresponds to different planes.

features among them.

3.3.1   Using two-views

Assuming that the multiple bodies are rigid and each body undergoes a different motion. Each distinct rigid-body motion (R,t) could be described by a corresponding fundamental matrix $F = [K't]_x K'RK^{-1}$ between the two views. That fundamental matrix satisfies the epipolar constrain $(p_r^T F p_l = 0)$ such that p_r and p_l are two matching features of a 3D point that belongs to the rigid body [18, 16].

We apply PEARL to estimate multiple fundamental matrices for uncalibrated image pairs. Matching pairs are found using the same procedure mentioned in section 3.2. One way to measure geometric error between a matching pair of points p and a given model F is the squared Sampsons distance (SSD) [13]

$$||p - F|| = \frac{\left(p_r^T F p_l\right)^2}{(F p_l)_1^2 + (F p_l)_2^2 + (F^T p_r)_1^2 + (F^T p_r)_2^2} \tag{9}$$

where the $(F p_l)_j^2$ represents the square of the j-th entry of the vector $(F p_l)$. We generate our finite set of initial model proposals F_0 by random sampling eight matching pairs. Then compute the model parameters as descried in [13] by minimizing the non-linear SSD error using Levenberg-Marquard. The initial solution for the non-linear minimization is found using the normalized 8-point algorithm [12]. The next step is to triangulate the features of one of the images (e.g the right image).Then we apply PEARL to energy

$$E(\mathbf{F}) = \sum_p ||p - F_p|| + \lambda \sum_{(p,q)\in\mathcal{N}} \delta(F_p \neq F_q) + \beta \cdot |\mathcal{F}|$$

where $\mathbf{F} = \{F | p \in P\}$ is an assignment of models to data points p, results are shown in Fig. 18.

### 3.3.2   Using multiple-views

Assuming that the cameras are affine then it could be proved that the motion of a rigid body i.e. the trajectory of its features will live in a 4D subspace [4, 22, 26]. Let P be a set of 3D points that belong to a rigid body which undergoes a motion over F frames. The image feature of a point i in the frame f is defined by p_{fi}. Stacking all p_{fi} measurements $\forall f \in F$ and $\forall i \in P$ will form the measurement
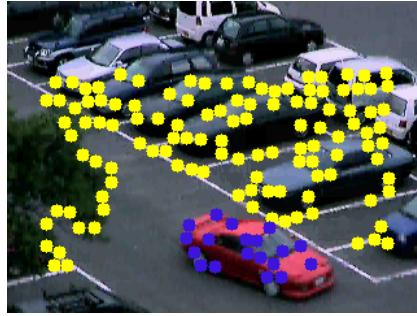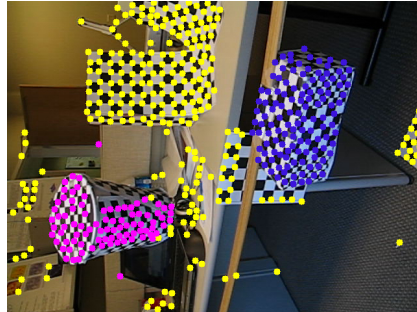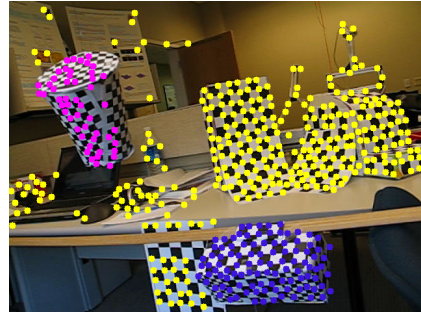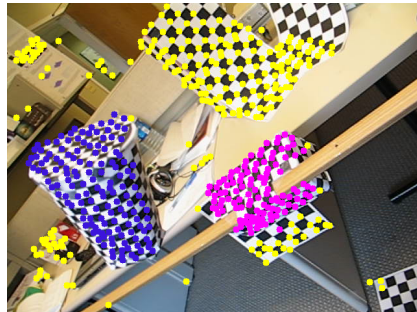
(a) kanatani1
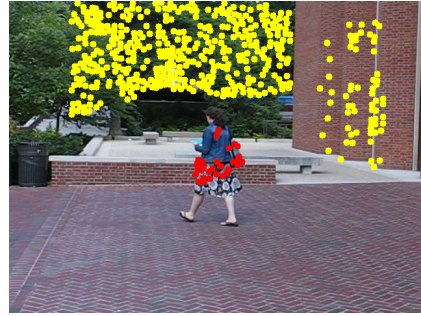
(b) people1
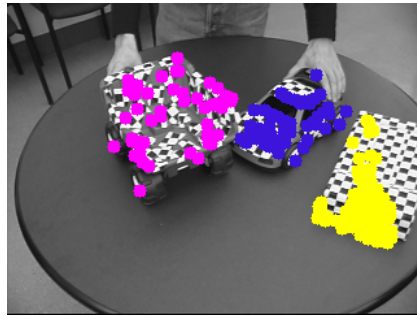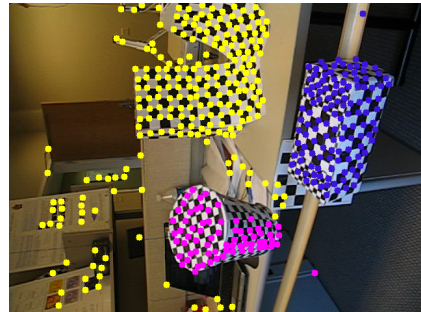
(c) three-cars

(d) people2

(e) 1R2RC

(f) 1R2RCR

(g) 1RT2RCR

(h) 1R2RCT_A

Figure 18: PEARL's results for eight different motion sequences from Rene Vidal's data set [25]. Our motion estimation based on the mixture of fundamental matrices uses only two frames (the first and the last).
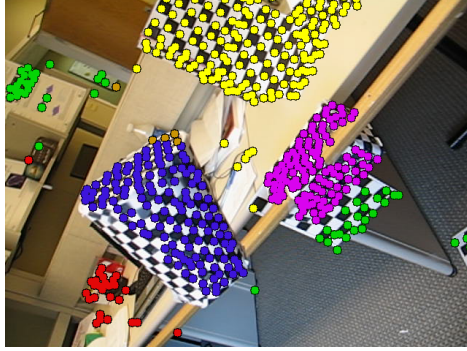
Figure 19: Optimal solution for energy (3). In contrast to results for energy (5) in Fig.18e-h, it fails to generate one background motion from yellow, green, and red clusters. These clusters correspond to infinitesimally close motions, but they are spatially isolated. The third term in (5) addresses this issue.

matrix

$$W = \begin{bmatrix} p_{11} & \cdots & p_{1P} \\ \vdots & & \vdots \\ p_{F1} & \cdots & p_{FP} \end{bmatrix}_{2F \times P} \tag{10}$$

which has rank 4 [4, 22].

First, we project these feature trajectories from $\mathcal{R}^{2F}$ to $\mathcal{R}^5$. The extra dimension is needed to discriminate between different motions [26]. Then we use PEARL to fit multiple 4D hyperplanes (motions). In contrast to [26], PEARL does not require the prior knowledge of the number of motions.

We generate our finite set of initial proposals M_0 by randomly sampling five points from the projected trajectories. Then we find the best fitting 4D hyperplane by minimizing the orthogonal distance $||p - M||$. The next step is to triangulate the 2D image features on one of the frames (e.g the last frame). Finally, we apply PEARL to the following energy

$$E(\mathbf{M}) = \sum_p ||p - M_p|| + \lambda \sum_{(p,q) \in \mathcal{N}} \delta(M_p \neq M_q) + \beta \cdot |\mathcal{M}|$$

where $\mathbf{M} = \{M | p \in P\}$ is an assignment of models to data points p, results are shown in Fig. 20.
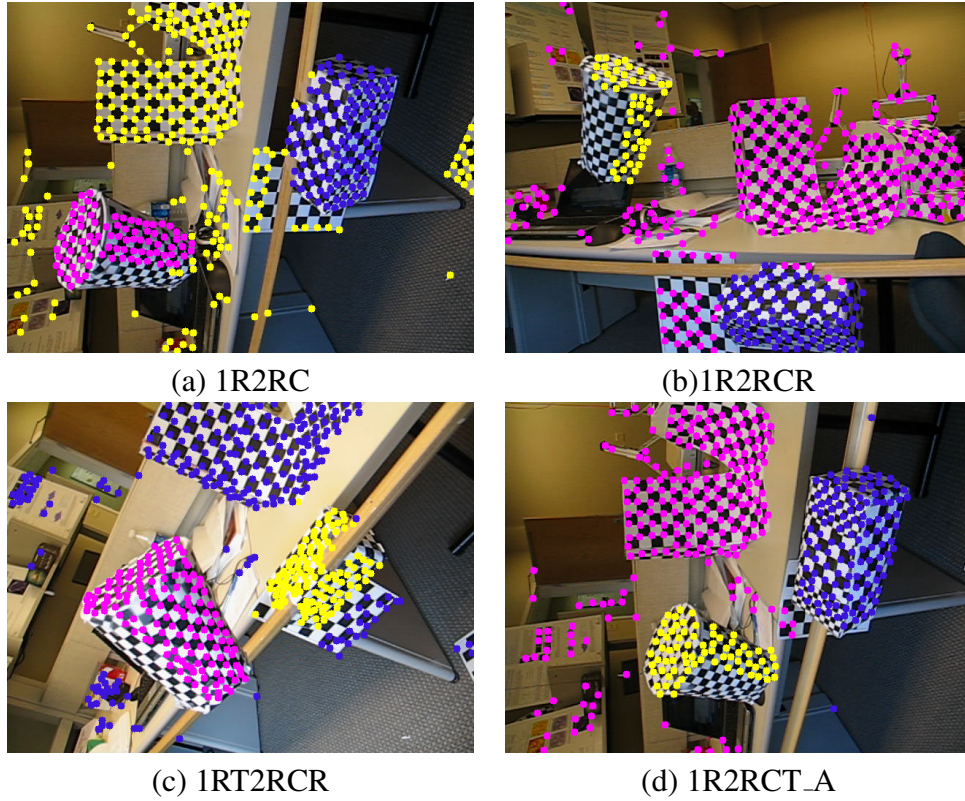
(a) 1R2RC

(b)1R2RCR

(c) 1RT2RCR

(d) 1R2RCT_A

Figure 20: These are PEARL's results for four different motion sequences from Rene Vidal's data set [25] based on multiple-views, see Sec.3.3.2. This method makes use of all frames in the motion sequence unlike the mixture of fundamental matrix method.

References

[1] J. S. Beis and D. G. Lowe. Shape indexing using approximate nearest-neighbour search in high-dimensional spaces. In *CVPR*, pages 1000–1006, 1997. 22

[2] S. Birchfield and C. Tomasi. Multiway cut for stereo and motion with slanted surfaces. In *ICCV*, 1999. 4, 8, 9, 23, 24, 25, 26

[3] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, August 2006. 5

[4] T. Boult and L. G. Brown. Factorization-based segmentation of motions. In *IEEE Workshop on Visual Motion*, 1991. 28, 30, 32

[5] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *PAMI*, 2001. 7, 11, 13, 15

[6] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *PAMI*, 2002. 3, 15

[7] J. Costeira and T. Kanade. A multi-body factorization method for motion analysis. In *ICCV*, 1995. 28

[8] A. Delong, A. Osokin, H. Isack, and Y. Boykov. Fast Approximate Energy Minization with Label Costs. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010. 9, 10, 11, 15, 21

[9] O. Faugeras and Q.-T. Luong. *The Geometry of Multiple Images*. MIT Press, 2004. 22

[10] M. A. Fischler and R. C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *CACM*, 1981. 2, 4, 11

[11] A. Gruber and Y. Weiss. Incorporating non-motion cues into 3D motion segmentation. In *European Conference on Computer Vision (ECCV)*, 2006. 4, 5

[12] R. Hartley. In defense of the eight-point algorithm. *PAMI*, 19(6):580–593, June 1997. 30

[13] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2003. 14, 27, 30

[14] H. Isack. *Spatially Coherent Multi-Model Fitting*. CS Dept., University of Western Ontario, London, Canada, April, 2009. 11

[15] Y. G. Leclerc. Constructing simple stable descriptions for image partitioning. *International Journal of Computer Vision (IJCV)*, 3(1):73–102, May 1989. 9

[16] H. Li. Two-view motion segmentation from linear programming relaxation. In *CVPR*, 2007. 4, 6, 8, 9, 10, 30

[17] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004. 22, 24

[18] Y. Ma, S. Soatto, J. Kosecka, and S. Sastry. *An Invitation to 3D Vision: From Images to Geometric Models*. Springer Verlag, 2003. 30

[19] M. Muja and D. G. Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. In *VISAPP*, 2009. 22

[20] C. Rother, V. Kolmogorov, and A. Blake. Grabcut - interactive foreground extraction using iterated graph cuts. In *ACM Transactions on Graphics (SIG-GRAPH)*, August 2004. 9

[21] R. Toldo and A. Fusiello. Robust multiple structures estimation with j-linkage. In *ECCV*, 2008. 3, 15

[22] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: a factorization method. *IJCV*, 1992. 28, 30, 32

[23] P. Torr and A. Zisserman. MLESAC: A New Robust Estimator with Application to Estimating Image Geometry. *Journal of Computer Vision and Image Understanding*, 78(1):138156, 2000. 4, 5

[24] P. H. S. Torr. Geometric Motion Segmentation and Model Selection. *Philosophical Trans. of the Royal Society A*, pages 1321–1340, 1998. 2, 4, 6, 8, 9, 10, 21

[25] R. Tron and R. Vidal. A benchmark for the comparison of 3-d motion segmentation algorithms. In *CVPR*, 2007. 31, 33

[26] R. Vidal, R. Tron, and R. Hartley. Multiframe motion segmentation with missing data using powerfactorization and gpca. *IJCV*, 2008. 30, 32

[27] E. Vincent and R. Laganiere. Detecting planar homographies in an image pair. In *ISPA*, June 2001. 2, 9, 11, 21

[28] R. Zabih and V. Kolmogorov. Spatially Coherent Clustering with Graph Cuts. In *CVPR*, June 2004. 9

[29] S. C. Zhu and A. Yuille. Region competition: Unifying snakes, region growing, and Bayes/MDL for multiband image segmentation. *IEEE Transactions*

on Pattern Analysis and Machine Intelligence, 18(9):884–900, September 1996. 9

[30] M. Zuliani, C. Kenney, and B. Manjunath. The multiransac algorithm and its application to detect planar homographies. In *ICIP*, 2005. 2, 11, 15, 21, 27