Branch-and-bound Hypothesis Selection for Two-view Multiple Structure and Motion Segmentation

Ninad Thakoor
Electrical Engineering Department,
University of Texas at Arlington, USA 76010,
Email:ninad.thakoor@uta.edu.

Jean Gao,
Computer Science and Engineering Department,
University of Texas at Arlington, USA 76010,
Email: gao@uta.edu.

Abstract

An efficient and robust framework for two-view multiple structure and motion segmentation is proposed. To handle this otherwise recursive problem, hypotheses for the models are generated by local sampling. Once these hypotheses are available, a model selection problem is formulated which takes into account the hypotheses likelihoods and model complexity. An explicit model for outliers is also added for robust model selection. The model selection criterion is optimized through branch-and-bound technique of combinatorial optimization which guaranties optimality over current set of hypotheses by efficient search of solution space.

1. Introduction

Segmentation of structure and motion is a vital step towards interpretation of a dynamic scene. A typical dynamic scene structure includes multiple independently moving objects which is being captured by a moving camera. Conventional approach based on frame difference [4] or 2-D flow based methods [11] are restricted in segmenting such a scene. Frame difference based approaches are limited due to requirement of camera motion compensation. While 2-D flow based approaches are limited by camera model used which is typically affine. To address the problem is a better way, a comprehensive theory of structure-and-motion (SaM) estimation from perspective images has been developed by computer vision researchers over the years [5]. Analysis of dynamic scenes based on this theory, also known as multi-body structure-and-motion (MSaM) is now being explored extensively. Two-view MSaM problem can be interpreted as a geometric problem [16]. However, its direct application in real world is limited as it lacks an outlier model. An interesting alternative is a clustering perspective. Two-view MSaM clustering is a chicken-and-egg problem. To segment the scene, one needs motion models for all the objects. To estimate the motion models of individual objects, one has to segment the objects first. To solve such a recursive problem, iterative techniques such as

expectation maximization can be used [3]. However, results of expectation maximization can be guaranteed to be only locally optimal and hence depend on the initialization. Alternative to iterative method is a sequential extraction strategy where dominant motions are segmented and separated one by one till entire scene is explained [7]. Limitation of such methods is that the similar motions are often incorrectly segmented. For objects having similar motions, sequential method assigns some fraction of other objects to the motion encountered first in its search.

To get out of chicken-and-egg dilemma, some researchers have applied Monte-Carlo sampling to generate multiple hypotheses for the motions in a scene [9, 17]. A prior knowledge of spatial coherency helps in selection of reliable hypotheses by local sampling though RANSAC like process. Once hypotheses are available through sampling, a suitable cost function can be optimized to achieve the motion segmentation. Another important aspect of clustering is the number of clusters itself. While most of the clustering techniques assume that the number of clusters is known, such assumption is invalid for the segmentation of a dynamic scene. Typically, clustering is carried out for varying number of clusters and the one fitting a certain criterion the best is selected. Under a sampling based framework, problem of selecting optimal number of clusters can be formulated as a combinatorial optimization problem.

Our paper gives a general combinatorial optimization framework to optimize a cost function integrating maximum likelihood of hypotheses, a clustering cost and uniform distribution of outliers. Initially, hypotheses for motion are generated by local sampling of matched features between two views. A null hypothesis is also introduced which suggests that any match can be an outlier with uniform likelihood. Next, a model selection criterion that penalizes the likelihood of the clustering with increasing number of clusters is added to the framework. The model selection criterion is optimized through branch-and-bound to obtain the final MSaM segmentation.

The paper is organized as follows: Section 2 formulates

the MSaM segmentation as a combinatorial optimization problem. A branch-and-bound solution to the problem is formulated in section 3. The experimental results are presented in section 4 and section 5 enlists the concluding remarks.

2. Formulation

Consider set of N image correspondences $\mathbf{X} = \{(\mathbf{x}_1, \mathbf{x}_1'), (\mathbf{x}_2, \mathbf{x}_2'), \ldots, (\mathbf{x}_N, \mathbf{x}_N')\}$, where $\mathbf{x}_i$ and $\mathbf{x}_i'$ are image coordinates of ith correspondence. Relationship of various object structures and motion in the scene can be expressed as,

$$\mathbf{x}_i'^T \left(\sum_{j=1}^{K} \mathcal{L}_j(i) \mathbf{F}_j \right) \mathbf{x}_i = 0. \qquad (1)$$

Here, $\mathbf{F}_j$ is the fundamental matrix for jth rigid body in the scene. The indicator function $\mathcal{L}_j(i)$ is one when $i$the correspondence belongs to the jth rigid body and zero otherwise. A label field $L = [l_1, l_2, \ldots, l_N]$ is associated with the indicator function $\mathcal{L}_j(i)$ such that, $l_i = j \leftrightarrow \mathcal{L}_j(i) = 1$. Goal of the MSaM segmentation is to estimate the label field L. Once the label field in known, fundamental matrix $\mathbf{F}_j$ can be computed as,

$$\mathbf{F}_j = \arg \min_{\mathbf{F}} \sum_{\forall i, l_i = j} d(\mathbf{x}_i, \mathbf{x}_i', \mathbf{F}). \qquad (2)$$

Here, d is a distance measure such as symmetric transfer error, reprojection error or Sampson approximation [5]. Under assumption of Gaussian errors above estimate is the optimal maximum likelihood estimate.

On the other hand, if $\mathbf{F}_j$s are known, the maximum likelihood estimate for the label of ith match is given by,

$$\hat{l}_i = \arg \min_{l} d(\mathbf{x}_i, \mathbf{x}_i', \mathbf{F}_l). \qquad (3)$$

Equations (2) and (3) are parameter estimation and label estimation or segmentation steps respectively. Since these steps are interdependent, the MSaM problem can be solved iteratively to maximize the likelihood of the correspondences. Log likelihood of the matches is given by,

$$\log\{\text{Lik}(\mathbf{X})\} = -\frac{N}{2} \log \left(\frac{\text{SSD}}{N} \right) + C, \qquad (4)$$

where,

$$\text{SSD} = \sum_{i=1}^{N} \min_{j=1}^{K} d(\mathbf{x}_i, \mathbf{x}_i', \mathbf{F}_j), \qquad (5)$$

and C is a constant. This optimization procedure also assumes that the number of clusters K is known *a priori*. This assumption is unrealistic in most of the scenes. Since likelihood of the correspondences increases if K is increased, likelihood alone cannot be applied to select optimal K. A model selection criterion such as Bayesian information criterion (BIC) or Akaike information criterion (AIC) can be

utilized to select the optimal K [10]. These criterions penalize likelihood in proportion of K. We define a generalized cost function to incorporate this idea,

$$\mathcal{C} = -2 \log\{\text{Lik}(\mathbf{X})\} + \alpha \cdot K. \qquad (6)$$

For BIC, $\alpha = \log(N)$ and for AIC, $\alpha = 2N$.

The cost function in (6) can be optimized by iterative optimization of likelihood in (4) for varying values of K. Finally, the optimal K can be selected to minimize $\mathcal{C}$. Alternative to this approach is simultaneous model selection and segmentation. In this approach, multiple hypotheses for fundamental matrix $\mathbf{F}_j$ are generated by using sampling on the correspondences. Once these hypotheses are know, MSaM segmentation problem is reduced to combinatorial optimization problem to select K hypotheses out of total N_h hypotheses. Note that there are 2^{N_h} possible solutions for this problem. Thus, even for moderate value of N_h, exhaustive search becomes intractable. However, nature of the problem allows us to use branch-and-bound approach to obtain optimal solution in reasonable time for practical problems.

3. Branch-and-bound

Branch-and-bound approach [1] to global optimization splits the optimization problem into smaller subproblems and for these subproblems, upper and/or lower bounds of cost function are estimated. These bounds are used to eliminate the subproblems that would not lead to an optimal solution. The subproblems that survive are further divided and the bound calculation is continued till all the subproblems are explored. Branch-and-bound procedure is popular in optimal feature subset selection [2] and is also used in clustering [8, 2] where number of clusters are known. Rest of this section constructs the branch-and-bound algorithm for the optimization of cost function in (6).

3.1. Solution tree

There are N_h hypotheses $H = \{\mathbf{F}_1, \mathbf{F}_2, \ldots, \mathbf{F}_{N_h}\}$ for the fundamental matrices $\mathbf{F}_j$ and we have to choose K of them to minimize the criterion in (6). All possible solutions of the optimization problem can be represented as a rooted tree. Each node encountered on the tree represents a solution. The node is also the partial solution for its descendent nodes. It is important that every solution is listed only once to avoid unnecessary computations. This can be ensured by creating child nodes that are different than *left siblings, ancestors* and *left siblings of ancestors*. One simple way of generating such a solution tree for $N_h = 5$ is shown in Figure 1 with an additional null hypothesis. We will introduce the null hypothesis later in the section. Note that, in the solution tree $(z_1 < z_2 < z_3 \ldots < z_n)$ and (left sibling $<$ right sibling). These two conditions ensure that the rule stated above to generate the child nodes is followed.
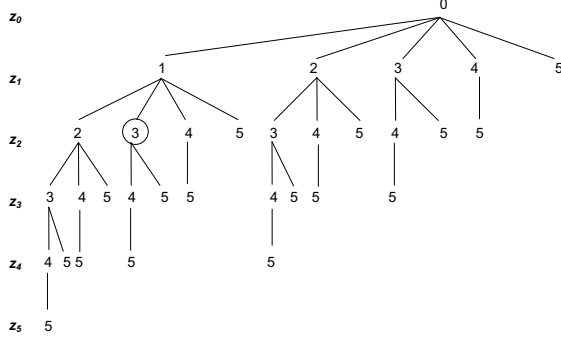
Figure 1. Solution tree for $N_h = 5$ and a null hypothesis

The solution tree is explored by series of branch forward, branch right and retraction operations. At the circled node in figure 1, the current partial solution is $\{0, 1, 3\}$. A branch forward operation moves deeper in the tree by one level. After branch forward operation, the partial solution $\{0, 1, 3\}$ would lead to $\{0, 1, 3, 4\}$. Branch right operation moves to the sibling branch towards right. Solution $\{0, 1, 3\}$ would branch right to give $\{0, 1, 4\}$. Retraction moves the solution one level up the tree. Retraction is carried out when no forward or right branching is possible. $\{0, 1\}$ is the result of the retraction at the circled node. The branch-and-bound algorithm is terminated when retraction leads to the root node.

3.2. Monotonocity of partial costs

A solution for a node at depth n be given by $Z(n) = \{(z_0, z_1, z_2, \ldots, z_{n-1}, z_n), \mathcal{D}(n)\}$. The solution $Z(n)$ represents n clusters which have cluster centers at $\{F_{z_0}, F_{z_1}, F_{z_2}, \ldots, F_{z_{n-1}}, F_{z_n}\}$. $\mathcal{D}(n) = \{D(1, n), D(2, n), \ldots, D(N, n)\}$ corresponds to minimum distances for the N matches under current cluster center hypothesis. Thus the cost function for this solution can be written as,

$$\mathcal{C}(Z(n)) = N \log \left\{ \frac{1}{N} \sum_{i=1}^{N} D(i, n) \right\} + \alpha \cdot n, \quad (7)$$

where,

$$D(i, n) = \min_{k=0}^{n} d(\mathbf{x}_i, \mathbf{x}'_i, F_{z_k}).$$

$D(i, n)$ gives the minimum distance for all the hypotheses included in solution till tree level n. When a new hypothesis is added to the existing partial solution it means that a new cluster center is being added. Matches which are close to the new cluster center are reassigned to the new cluster while others remain unchanged. In terms of the partial solution $D(i, n-1)$, the newly formed $D(i, n)$ can be calculated incrementally as,

$$D(i, n) = \min[D(i, n - 1), d(\mathbf{x}_i, \mathbf{x}'_i, F_{z_n})]. \quad (8)$$

From (8), $D(i, n)$ has to be lesser than or equal to $D(i, n-1)$. The cost function in (6) is made up of two terms, one

corresponding to the inverse likelihood and another corresponding to the number of clusters. When a new cluster center is added, the second term of cost function increases by α while inverse likelihood term decreases. We will use this monotonocity property in the following subsection to establish the lower bound on the cost function.

Leading from the monotonic decrease of the inverse likelihood and linear increase of clustering cost, a monotonicity requirement can be imposed on the optimal solution. According to this requirement, for an optimal solution $\{z_0, z_1, \ldots, z_{m-1}, z_m, z_{m+1}, \ldots, z_n\}$, we must have:

$$\mathcal{C}(z_0, z_1, \ldots, z_{m-1}, z_m) < \mathcal{C}(z_0, z_1, \ldots, z_{m-1})$$

for any value $m \leq n$. If decrease in negative likelihood from $Z(m-1)$ to $Z(m)$ is given by Δ_m, then for monotonic decrease of cost, $\Delta_m > \alpha$. For any m which has $\Delta_m < \alpha$, a better solution can be achieved by removing z_m from current solution as there will be decrease of at least $\alpha - \Delta_m$ in the cost. Elimination of z_m corresponds to branch right operation in the solution tree which abandons solutions stemming from z_m.

3.3. Lower bound on cost

To establish lower bound on the cost, we define a complementary variable $D^*(i, z_n)$ as,

$$D^*(i, z_n) = \min_{k=z_n+1}^{N_h} d(\mathbf{x}_i, \mathbf{x}'_i, F_k).$$

The variable $D^*(i, z_n)$ gives the minimum of the distance measure from all hypotheses which can be included in the solution in future. In case of the variable $D^*(i, z_n)$, its value solely depends on last node z_n. As there are only N_h possibilities for value of z_n, $D^*(i, z_n)$ can be pre-computed to speed up the branch-and-bound process. Similar to $D(i, n)$, $D^*(i, n)$ can also be calculated incrementally as,

$$D^*(i, z_n) = \min[D^*(i, z_n + 1), d(\mathbf{x}_i, \mathbf{x}'_i, F_{z_n+1})].$$

Consider a possible partial solution $Z(4) = \{0, 1, 3, 4, 7\}$ for $N_h = 10$. The variable D at level n can be computed as,

$$D(i, 4) = \min\{d(\mathbf{x}_i, \mathbf{x}'_i, F_0), d(\mathbf{x}_i, \mathbf{x}'_i, F_1),$$
$$d(\mathbf{x}_i, \mathbf{x}'_i, F_3), d(\mathbf{x}_i, \mathbf{x}'_i, F_4), d(\mathbf{x}_i, \mathbf{x}'_i, F_7)\}.$$

Now for the same example, we consider the complementary variable D^*.

$$D^*(i, F_7) = \min\{d(\mathbf{x}_i, \mathbf{x}'_i, F_8), d(\mathbf{x}_i, \mathbf{x}'_i, F_9), d(\mathbf{x}_i, \mathbf{x}'_i, F_{10})\}.$$

With help of the complementary variable, lower bound on the possible solution leading from $Z(n)$ is,

$$\mathcal{C}_{\text{Lower}}(Z(n)) = N \log \left\{ \frac{1}{N} \sum_{i=1}^{N} \min[D(i, n), D^*(i, z_n)] \right\}$$
$$+ \alpha \cdot (n + 1). \quad (9)$$

If $\mathcal{C}_{\text{Lower}}(Z(n)) > \mathcal{C}^*$, then the current partial solution can be safely abandoned as it would not lead to better solution than current optimal solution $\mathcal{C}^*$.

3.4. Outlier likelihood

Matching errors are common in MSaM problems. These errors can severely deteriorate quality of the solutions achieved for MSaM segmentation. Outliers can be assumed to be uniformly distributed throughout the image with likelihood d_0. For ease of notation, we assume that,

$$\log(d_0) = d(\mathbf{x}_i, \mathbf{x}'_i, \mathbb{F}_0).$$

With introduction of this outlier likelihood, the proposed MSaM segmentation scheme would act as a simple redescending M-estimator [6].

3.5. Branch-and-bound algorithm

Based on the monotonocity requirement and the lower bound, the branch-and-bound segmentation algorithm is listed below.

1. Initialization: Set the tree level $n = 1$, current node $z_0 = 0$ and current optimal cost $\mathcal{C}^* = \mathcal{C}(Z(0))$.

2. Generate child nodes: Initialize $LIST(n)$,

$$List(n) = \{z_{n-1} + 1, z_{n-1} + 2, \dots, N_h\}$$

3. Select new node: If $List(i)$ is empty, to step (5). Otherwise, set $z_n = k$ where $k \in List(i)$. Set current solution $Z(n) = \{z_0, z_1, \dots, z_n\}$. Delete k from $List(i)$.

4. Check bounds:

   - Compute $\mathcal{C}(Z(n))$ and $\mathcal{C}_{\text{Lower}}(Z(n))$.
   - If $\mathcal{C}(Z(n)) < \mathcal{C}^*$, set $\mathcal{C}^* = \mathcal{C}(Z(n))$ and $Z^* = Z(n)$.
   - If $\mathcal{C}(Z(n-1)) < \mathcal{C}(Z(n))$ or $\mathcal{C}_{\text{Lower}}(Z(n)) > \mathcal{C}^*$, go to step (3).
   - If $\mathcal{C}(Z(n-1)) > \mathcal{C}(Z(n))$ and $\mathcal{C}_{\text{Lower}}(Z(n)) < \mathcal{C}^*$, set $n = n + 1$ and go to step (2).

5. Backtrack to lower level: Set $n = n - 1$. if $n > 0$ go to step (3), otherwise terminate the algorithm.

In the following section, the branch-and-bound hypothesis selection was implemented and results achieved are resented.

4. Experimental results

The proposed MSaM segmentation approach was implemented and tested with publicly available data sets. MSaM segmentation was implemented in Matlab and executed on a Core 2 Duo processor operating at 2.33GHz as a single thread.

To generate the motion hypotheses, for each matched image feature fundamental matrix was computed from its
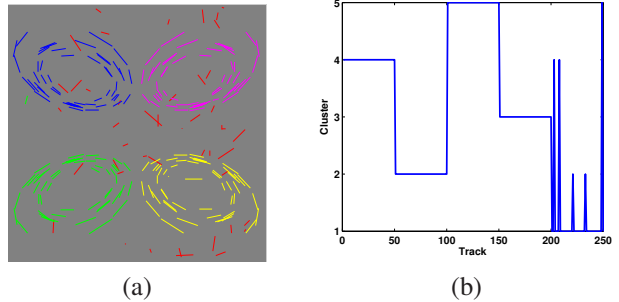

(a)　　　　　　　　　　　(b)

Figure 2. Spinning wheels: (a) Segmentation results, 4 motions detected, red tracks are outliers; (b) Memberships of various tracks, 201 to 250 are outliers.

neighborhood. Due to sparse nature of data, neighborhood was generated through Delaunay triangulation. Matches in the neighborhood were used to compute fundamental matrix using "Structure and Motion Toolkit" from [15]. Similar to RANSAC, outliers and inliers were selected for each fundamental matrix with a threshold. To avoid repeated hypothesis which are similar, non maximal suppression was carried out for the matches based on number of inliers. Finally, the surviving hypotheses were arranged in decreasing order of number of inliers. Bayesian information criterion (BIC) was optimized for these hypotheses to select optimal hypothesis.

In the first experiment, we use "Spinning wheels" synthetic test data from [13]. This sequence contains four rotating objects with 50 tracked points each with 50 outliers. Frame 1 and 3 of the sequence were used in our experiment. After sampling and non maximal suppression, 27 hypotheses were selected. Proposed approach detects 4 clusters along with outliers. All the inliers were correctly labeled while 5 of the outliers were mislabeled. Total number of solutions explored by the branch-and-bound was 139.

For the second experiment, "Box-book-mag" and "Desk" image pairs from [12] are used. "Box-book-mag" pair has three independently moving objects while camera is stationary. Figure 3 (a) shows disparities between the image pair indicated in different colors. While red colored matches are detected outliers, each of rest of the colors represents disparities for a segmented object. For the "Desk" image pair shown in figure 4, there are three moving objects namely pile of books, a computer screen and a journal. Although the camera has also moved, there are no matches for the background. Thus the background motion is not detected. Result of segmentation can be seen in figure 4 (b) and (c).

In the next experiment, our method is applied to the "car-truck-box" sequence used by Vidal et al. [18]. Motion between frame 1 and frame 4 of the sequence was analyzed. In this sequence, there are three different motions. The box lies on a rotating desk, while car and truck are moved away
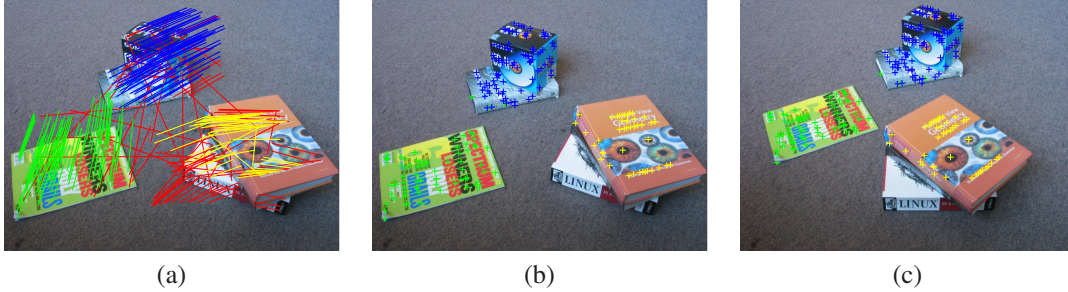
Figure 3. Box-book-mag: (a) Disparities between two images, each cluster is denoted by different color, matches marked by red are outliers; (b) Segmentation result for the first image; (c) Segmentation result for the second image.
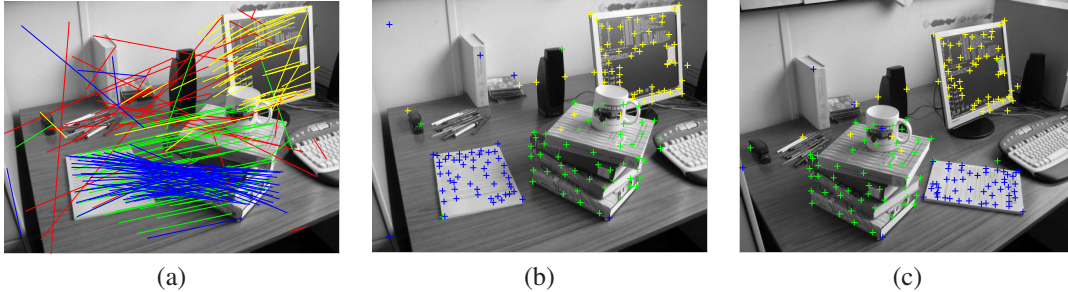

Figure 4. Desk: (a) Disparities between two images, each cluster is denoted by different color, matches marked by red are outliers; (b) Segmentation result for the first image; (c) Segmentation result for the second image.

Table 1. Execution summary for the experiments

Experiment	N_h	Solutions explored	Fraction explored	Search time (ms)
Spinning wheels	27	129	1.03e-006	391
Box-book-mag	31	701	3.26e-007	94
Desk	20	535	5.10e-004	78
Car-truck-box	19	73	1.39e-004	47
Kanatani	32	116	2.70e-008	32

from each other with hand. However, these motions are very small and are magnified for plotting purpose in figure 5(a). Three moving objects are correctly identified; however some of the motion vectors are incorrectly assigned. This is due to the sampling scheme that we use, rather than the cost function being optimized. If optimal motions are subset of the hypotheses being constructed then segmentation results are guaranteed to optimal.

In the final sequence, taken from Sugaya and Kanatani [14] has a single moving object, i.e. the car. However, camera is also moving for this sequence. Frame 1 and frame 11 are used for segmentation in our experiment. Egomotion of camera and motion of the car are correctly segmented and are shown in figure 6 (c).

Table 1 shows summary of the execution of our method for all the experiments. Fraction of solutions explored shown in the table is calculated as,

$$\text{Fraction explored} = \frac{\text{Solution explored}}{2^{N_h}}.$$

As seen for the table, fraction of all the solutions explored is very small. This is also reflected in the execution speed.

Note that execution times for search alone are listed and they do not include sampling and pre-computing involved. Speedups achieved increase with increase in N_h since more solutions are generally rejected implicitly by explicitly rejecting a partial solution.

5. Concluding remarks

We have proposed a versatile multiple structure and motion segmentation scheme and demonstrated its effectiveness through experiments. The branch-and-bound scheme can easily be scaled for parallel processing by solving one branch of the problem on a processor. Scheduling of these branches can be also an interesting direction of research. Although the method is proposed for a multi body SaM segmentation, it can be also applied to various other computer vision problems involving clustering such as segment based stereo and motion segmentation. Since the outcome of the method heavily depends on the initial hypotheses chosen, various available guided sampling approaches have to be evaluated as to how well they explore and represent the solution space. Current approach can also be extended to an iterative approach. After each iteration of segmentation, fundamental matrices can be calculated based on membership of the matches and these can added as additional hypothesis to repeat the segmentation.

References

[1] M. Brusco and S. Stahl. *Branch-and-Bound Applications in Combinatorial Data Analysis*. Springer, 2005.
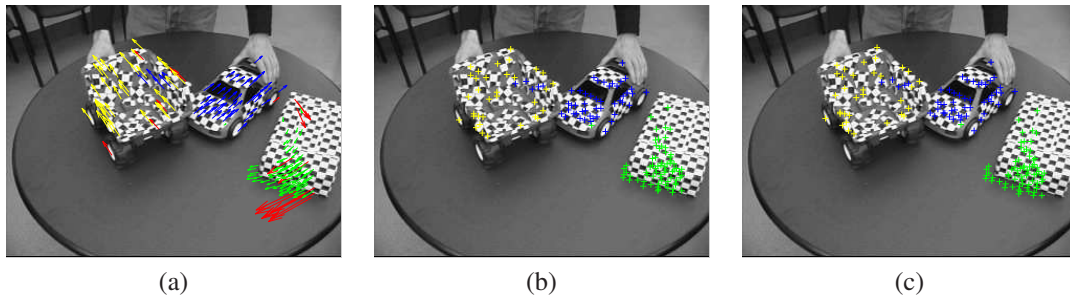
Figure 5. Car-truck-box: (a) Disparities between two images, each cluster is denoted by different color, matches marked by red are outliers; (b) Segmentation result for frame 1; (c) Segmentation result for frame 4.
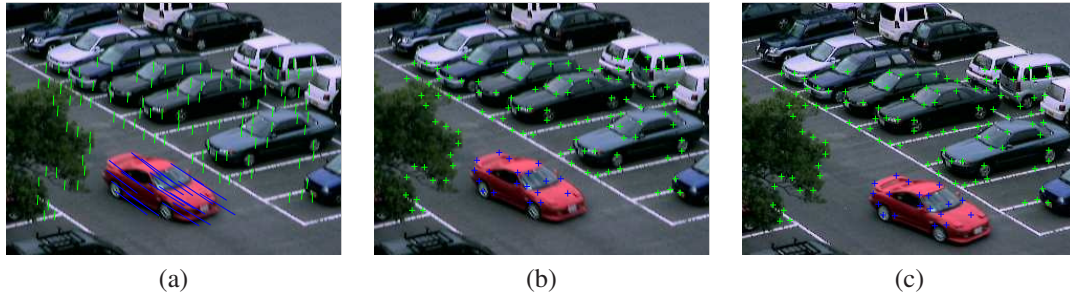

Figure 6. Kanatani: (a) Disparities between two images, each cluster is denoted by different color; (b) Segmentation result for frame 1; (c) Segmentation result for frame 11.

[2] K. Fukunaga. *Introduction to statistical pattern recognition (2nd ed.)*. Academic Press Professional, Inc., San Diego, CA, USA, 1990.

[3] A. Gruber and Y. Weiss. Incorporating non-motion cues into 3d motion segmentation. In *Proceedings of ECCV*, pages 84–97, 2006.

[4] I. Haritaoglu, D. Harwood, and L. S. Davis. W4: real-time surveillance of people and their activities. *IEEE Trans. Patt. Anal. Mach. Intell.*, 22(8):809–830, Aug 2000.

[5] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[6] P. J. Huber. *Robust statistics*. Wiley, 1981.

[7] M. Irani and P. Anandan. A unified approach to moving object detection in 2d and 3d scenes. *IEEE Trans. Patt. Anal. Mach. Intell.*, 20(6):577–589, 1998.

[8] W. L. G. Koontz, P. M. Narendra, and K. Fukunaga. A branch and bound clustering algorithm. *IEEE Trans. Comput.*, 24(9):908–915, 1975.

[9] H. Li. Two-view motion segmentation from linear programming relaxation. In *Proceedings of CVPR*, pages 1–8, June 2007.

[10] A. D. R. McQuarrie and C.-L. Tsai. *Regression and Time Series Model Selection*. World Scientific, 1998.

[11] H. Nguyen, M. Worring, and A. Dev. Detection of moving objects in video using a robust motion similarity measure. *IEEE Trans. Img. Process.*, 9(1):137–141, Jan 2000.

[12] K. Schindler and D. Suter. Two-view multibody structure-and-motion with outliers through model selection. *IEEE Trans. Patt. Anal. Mach. Intell.*, 28(6):983–995, 2006.

[13] K. Schindler, J. U, and H. Wang. Perspective n-view multibody structure-and-motion through model selection. In *Proceedings of ECCV*, pages I: 606–619, 2006.

[14] Y. Sugaya and K. Kanatani. Multi-stage optimization for multi-body motion segmentation. *IEICE Trans. Inf. & Syst.*, E87-D(7):1935–1942, July 2004.

[15] P. Torr. *A Structure and Motion Toolkit in Matlab*. http://cms.brookes.ac.uk/staff/PhilipTorr/Beta/torrsam.zip.

[16] R. Vidal and Y. Ma. A unified algebraic approach to 2-d and 3-d motion segmentation. In *Proceedings of ECCV*, pages 1–15, 2004.

[17] R. Vidal and S. Sastry. Optimal segmentation of dynamic scenes from two perspective views. In *Proceedings of CVPR*, volume 2, pages 281–286, June 2003.

[18] R. Vidal, S. Soatto, Y. Ma, and S. Sastry. Segmentation of dynamic scenes from the multibody fundamental matrix. In *ECCV Workshop on Visual Modeling of Dynamic Scenes*, 2002.