Simultaneous Multi-Body Stereo and Segmentation

Guofeng Zhang[1] Jiaya Jia[2] Hujun Bao[1]

[1]State Key Lab of CAD&CG, Zhejiang University
{zhangguofeng, bao}@cad.zju.edu.cn

[2]The Chinese University of Hong Kong
leojia@cse.cuhk.edu.hk

Abstract

This paper presents a novel multi-body multi-view stereo method to simultaneously recover dense depth maps and perform segmentation with the input of a monocular image sequence. Unlike traditional multi-view stereo approaches that generally handle a single static scene or an object, we show that depth estimation and segmentation can be jointly modeled and be globally solved in an energy minimization framework for ubiquitous scenes containing multiple independently moving rigid objects. Our major contribution includes a new multi-body stereo model, which integrates the color, geometry, and layer constraints for spatio-temporal depth recovery and automatic object segmentation. A two-pass optimization scheme is proposed to progressively update the estimates. Our method is applied to a variety of challenging examples.

1. Introduction

Both stereo-based 3D reconstruction and image/video segmentation have been fundamental problems in computer vision for long time, due to the critical need of high quality depth and segment estimates in many applications, e.g., recognition, image-based rendering, and image/video editing. However, these two problems were researched typically along different lines.

In multi-view stereo [16], which estimates depth and 3D geometry from a collection of images, simultaneous dense 3D reconstruction and segmentation of rigid objects that move differently is very difficult. Coarse representation with multiple rigid components [14], 3D motion segmentation to separate feature trajectories of multiple moving objects [4, 21, 13], and object recognition with a training process [24, 9] were proposed to deal with dynamic or static scenes. They however cannot solve the high-quality dense 3D reconstruction problem, especially when moving objects are not initially separated.

In this paper, we present a new method to simultaneously achieve dense depth estimation and motion segmentation for multiple rigid objects undergoing different movements. Our major contributions include a new multi-body stereo representation that couples depth and segmentation labels, and a global estimation method to minimize a unified objective function, which notably extends multi-view stereo to scenes with several surfaces independent in motion. We also propose an adaptive-frame-selection scheme with a depth and segment hole filling algorithm for effective occlusion handling. The objective function is solved by an iterative optimization scheme. It first initializes labels with a novel multi-body plane fitting algorithm, and then iteratively refines them by incorporating the geometry and segment coherence constraints in a statistical way among multiple frames. Our method can yield spatio-temporally consistent depth and segment maps.

Previous Work and Discussion

3D motion segmentation separates feature trajectories of moving objects to recover their positions and the corresponding camera motion. Most of these methods adopt the affine camera model for simplification [4, 21, 13]. A few also aim to handle multiple perspective views [15, 12]. These approaches do not aim at high-quality dense 3D reconstruction with segmentation.

In 2D motion segmentation [1, 23, 29, 8], pixels that undergo similar motion are approximately grouped, and are separated into layers. These methods also depend on the accuracy of motion estimation and generally decouple the computation of motion and segmentation, which could introduce the *'chicken and egg'* problem – that is, inaccurate motion estimate causes segmentation ambiguity, while erroneous segments may adversely affect motion estimation.

Rothganger *et al.* [14] proposed reconstructing groups of affine-covariant scene patches with the multi-view constraints. It only coarsely represents a dynamic scene with multiple rigid components. Two recent methods [24, 9] performed semantic scene parsing and object recognition based on estimated dense depth maps, or by a joint optimization of segmentation and stereo reconstruction. These methods require a training stage and the scene must be static. In addition, the produced coarse object segments may be with

imprecise boundaries.

If moving rigid objects are masked out, we can apply MVS to each object independently. State-of-the-art segmentation methods, such as mean shift [3], normalized cuts [18], and weighted aggregation (SWA) [17] base their operations on 2D image structures and do not consider rich geometry in MVS.

With the objective to accurately extract foreground moving objects with visually plausible boundaries, *bilayer* segmentation methods [5, 19] were proposed assuming that the camera is mostly stationary, availing estimating or modeling the background color. Obviously, these methods, due to the static camera constraint, do not suit MVS either.

Recently, Zhang *et al*. [26] used both the motion and depth information to model the background scene and extracted good-quality foreground layer. The estimated dense motion field and bilayer segmentation are iteratively refined. This approach is limited to bilayer segmentation. In addition, only the motion field for the foreground layer is computed, which is not enough for 3D reconstruction.

2. System Overview

We first define notations used in this paper. Given a sequence $\hat{I}$ with n frames, i.e., $\hat{I} = \{I_t | t = 1, ..., n\}$, taken by a freely moving camera, our objective is to estimate the disparity maps $\hat{D} = \{D_t | t = 1, ..., n\}$ in the n frames as well as the corresponding motion segment maps $\hat{S} = \{S_t | t = 1, ..., n\}$. $I_t(\mathbf{x})$ denotes the color (or intensity) of pixel $\mathbf{x}$ in frame t.

We denote by K the number of independently moving rigid objects. If pixel $\mathbf{x}$ is in the kth object, we set $S_t(\mathbf{x}) = k$. Denoting by $z_{\mathbf{x}}$ the depth of pixel $\mathbf{x}$ in frame t, by convention, disparity $D_t(\mathbf{x})$ is defined as $D_t(\mathbf{x}) = 1/z_{\mathbf{x}}$.

2.1. Multi-Body Structure-from-Motion

In a conventional static-scene sequence, only one set of camera parameters is computed for each frame. Here, since we have K independently moving rigid objects, they have their own motion parameters and are viewed from different positions. The camera parameters of object k in frame t are denoted as $\mathbf{C}_t^k = \{\mathbf{K}_t, \mathbf{R}_t^k, \mathbf{T}_t^k\}$, where $\mathbf{K}_t$ is the intrinsic matrix, which is the same for all objects. $\mathbf{R}_t^k$ is the rotation matrix, and $\mathbf{T}_t^k$ is the translation vector for object k.

In this paper, with the focus to solve for dense 3D motion segmentation, the number K of rigid objects and the relative camera motion for each object are empirically computed by the multi-body structure-from-motion (SFM) method [12] in a pre-process. When occasional error arises in this automatic method due to complex structures of the sequence or the large number of independently moving objects, we remove problematic feature tracks, and use the semi-automatic method [2] to add a few long tracks, which are then manually grouped with respect to objects. One example is shown in Figure 1, where features are tracked for the two boxes. We perform structure-from-motion [28] for each group of the feature tracks independently such that relative camera motion can be respectively estimated for the objects. We sort the objects according to their distance to the camera. The relative scales among different objects are not estimated since the objects are generally not in contact and scales do not influence the depth estimation and segmentation.

(a) (b)

Figure 1. Pre-processing. (a-b) The grouped feature tracks for the two boxes in two selected frames. The tracked features in different objects are shown as green and red crosses, respectively. The white curves are the corresponding temporal trajectories.

After pre-processing, we estimate dense depth and segmentation maps with the multi-body configuration. It is challenging even for manual labeling of the layers that include fine details in each frame and of dense disparity values. So a robust automatic algorithm is needed.

2.2. The Framework

Table 1 gives an overview of our system. With an input sequence and the estimated camera motion for the objects, we first initialize the depth and object segmentation maps for each frame without temporal consideration. A new multi-body plane fitting scheme is introduced. Then we update the disparity and segmentation maps with iterative optimization. Finally, a hierarchical belief propagation algorithm is employed to densify the levels of disparity for higher estimation precision.

1. **Initialization**:
   - 1.1 Initialize depth and motion segmentation for each frame by solving Eq. (11) (Sec. 4).
   - 1.2 Use multi-body plane fitting to refine initialization (Sec. 4.1).
2. **Iterative Optimization**:
   - 2.1 Process frames consecutively from 1 to n: For each frame t, fix the disparities and segmentation labels in other frames and refine L_t by minimizing Eq. (1) (Sec. 4.2).
   - 2.2 Repeat step 2.1 for two passes.
   - 2.3 Use a hierarchical BP algorithm to increase estimation accuracy.

Table 1. Our Framework

3. Multi-Body Stereo Model

For each pixel, our goal is not only to estimate its actual disparity value, but to determine the object segment it belongs to as well. To this end, for object k, we first determine its maximum and minimum depth values of the recovered 3D points corresponding to the tracked features in multi-body structure-from-motion, and denote them as $z^k_{\max}$ and $z^k_{\min}$. The range of disparities is thus $[d^k_{\min}, d^k_{\max}]$ where

$$d^k_{\min} = s_{\min}/z^k_{\max}, \qquad d^k_{\max} = s_{\max}/z^k_{\min}.$$

The two scale factors $s_{\min} < 1$ and $s_{\max} > 1$. The disparity range is then evenly partitioned into m_k levels with interval Δd, such that the ith level is expressed as

$$d^k_i = (i-1)\Delta d + d^k_{\min},$$

where $i = 1, ..., m_k$. Now each pixel $\mathbf{x}$ has two key variables to be estimated: one is its disparity value d and the other is the segment index k. Separately computing these two sets of variables, as aforementioned, is not optimal and easily accumulates errors.

We alternatively propose an *expanded labeling set* that jointly considers these two variables for each pixel, and define it as

$$\mathcal{L} = \{d^1_1, d^1_2, ..., d^1_{m_1}, ..., d^K_1, d^K_2, ..., d^K_{m_K}\}.$$

The cardinality of the set $|\mathcal{L}| = \sum_{k=1}^{K} m_k$. In $\mathcal{L}$, each label (denoted as $\mathcal{L}_i$ for the ith label) naturally encodes a segment index and the actual disparity value. If a pixel is labeled as $\mathcal{L}_i$ after computation, we can easily determine its segment index $\mathcal{S}(\mathcal{L}_i)$ as

$$\mathcal{S}(\mathcal{L}_i) = h \quad \text{s.t.} \quad 1 \le i - \sum_{j=1}^{h-1} m_j \le m_h.$$

Its disparity value $\mathcal{D}(\mathcal{L}_i)$ is accordingly

$$\mathcal{D}(\mathcal{L}_i) = d^h_{i-\sum_{j=1}^{h-1} m_j}.$$

For example, $\mathcal{L}_{m_1+3}$ means that this pixel belongs to 2nd object, and the disparity value is d^2_3.

Thanks to this compact representation, instead of estimating D_t and S_t separately, we now can estimate a joint label map L_t for each frame t with the consideration of necessary color and geometry constraints.

3.1. Objective Function

To compute the label maps L for all frames, we define the energy in the input sequence as

$$E(L; \hat{I}) = \sum_{t=1}^{n} (E_d(L_t) + E_s(L_t)), \tag{1}$$

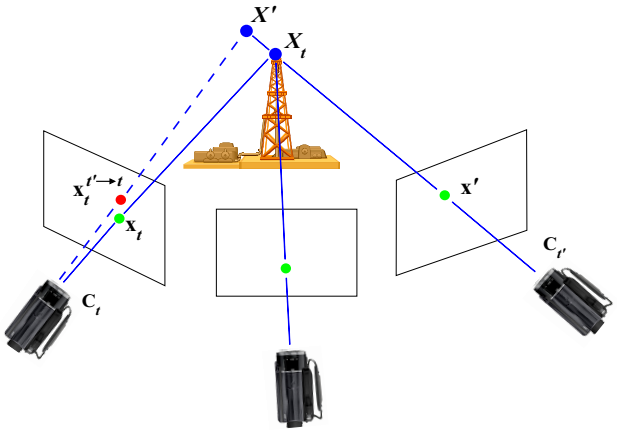

Figure 2. Multi-view geometry. Given disparity $\mathcal{D}(l)$, pixel $\mathbf{x}_t$ is mapped to its actual 3D position, and then reprojected to frame t'. The projected pixel in frame t' is denoted as $\mathbf{x}'$. Ideally, when we reproject $\mathbf{x}'$ from frame t' back to t, the projected pixel $\mathbf{x}_t^{t'\to t}$ should be identical to $\mathbf{x}_t$. In practice, due to matching errors, $\mathbf{x}_t^{t'\to t}$ and $\mathbf{x}_t$ are possibly different points.

where the data term E_d measures how well labeling L fits the observation $\hat{I}$, and the term E_s encodes spatial labeling smoothness. We elaborate these terms below followed by description of optimization and system initialization, and by the discussion of other implementation issues.

3.2. Data Term

Our data term takes the intensity, disparity, and layer consistency information into consideration. The likelihood that one pixel $\mathbf{x}_t$ in I_t is labeled as $l \in \mathcal{L}$ is defined as

$$P(\mathbf{x}_t, l) = \frac{1}{|\phi_v(\mathbf{x}_t)| + |\phi_o(\mathbf{x}_t)|}\Big(\sum_{t' \in \phi_o(\mathbf{x}_t)} p_o(\mathbf{x}_t, l, L_{t',t}) + \sum_{t' \in \phi_v(\mathbf{x}_t)} p_c(\mathbf{x}_t, l, I_t, I_{t'}) \cdot p_v(\mathbf{x}_t, l, L_{t'})\Big), \tag{2}$$

where $\phi_v(\mathbf{x}_t)$ and $\phi_o(\mathbf{x}_t)$ are two sets of the selected neighboring frames for $\mathbf{x}_t$, and $p_o(\mathbf{x}_t, l, L_{t',t})$ is a labeling prior, all of which will be elaborated in Section 3.3. $1/(|\phi_v(\mathbf{x}_t)| + |\phi_o(\mathbf{x}_t)|)$ is used for energy normalization. $p_c(\mathbf{x}_t, l, I_t, I_{t'})$ measures the color similarity between pixel $\mathbf{x}_t$ and the projected $\mathbf{x}'$ in frame t', same as the one in [27]:

$$p_c(\mathbf{x}_t, l, I_t, I_{t'}) = \frac{\sigma_c}{\sigma_c + ||I_t(\mathbf{x}_t) - I_{t'}(\mathbf{x}')||}, \tag{3}$$

where σ_c controls the shape of the differentiable robust function. $I_{t'}(\mathbf{x}')$ is the color of pixel $\mathbf{x}'$. With the estimated camera parameters and disparity $\mathcal{D}(l)$ of pixel $\mathbf{x}_t$, the location of the projected pixel $\mathbf{x}'$ can be expressed as

$$\mathbf{x}'^h \sim \mathbf{K}_{t'}\mathbf{R}_{t'}^\top \mathbf{R}_t \mathbf{K}_t^{-1}\mathbf{x}_t^h + \mathcal{D}(l)\mathbf{K}_{t'}\mathbf{R}_{t'}^\top(\mathbf{T}_t - \mathbf{T}_{t'}), \tag{4}$$

where the superscript h indicates the homogeneous coordinate of the vector. The 2D point $\mathbf{x}'$ is computed by dividing $\mathbf{x}'^h$ with the third homogeneous coordinate.

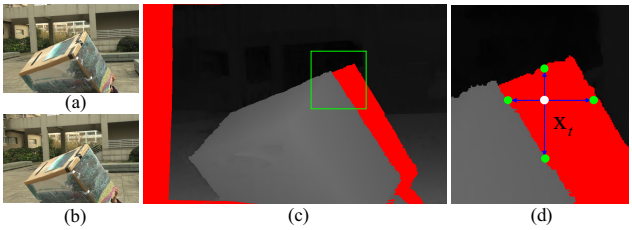

Figure 3. The projected labeling prior with hole filling. (a) The 31st frame. (b) The 36th frame. (c) The projected labeling prior $L_{31,36}$. The red pixels are those receiving no projection during the 3D warping. (d) Label inference for pixel $\mathbf{x}_t$ from the four nearest visible neighbors horizontally and vertically.

$p_v(\mathbf{x}_t, l, L_{t'})$ is a geometry and segment coherence term measuring whether or not pixel $\mathbf{x}_t$ and the projected correspondence $\mathbf{x}'$ are in the same object segment, and how consistent they are in terms of multi-view geometry. We define $p_v(\cdot)$ as

$$p_v(\mathbf{x}_t, l, L_{t'}) = \begin{cases} 0, & \mathcal{S}(l) \neq \mathcal{S}(l') \\ p_g(\mathbf{x}_t, \mathcal{D}(l), \mathcal{D}(l')), & \mathcal{S}(l) = \mathcal{S}(l') \end{cases} \quad (5)$$

where $l' \in \mathcal{L}$ is the current label of $\mathbf{x}'$. Eq. (5) shows if l and l' have different segment indices, the two pixels are not corresponding in the two frames and should be disconnected. Otherwise, we use p_g defined below to measure the geometric coherence between $\mathbf{x}_t$ and $\mathbf{x}'$ [27]:

$$p_g(\mathbf{x}_t, \mathcal{D}(l), \mathcal{D}(l')) = \exp(-\frac{||\mathbf{x}_t - \mathbf{x}_t^{t' \to t}||^2}{2\sigma_d^2}), \quad (6)$$

where $\mathbf{x}_t^{t' \to t}$ is the corresponding point in frame t by projecting $\mathbf{x}'$ from frame t' to t with its disparity estimate $\mathcal{D}(l')$. An illustration is provided in Figure 2. The standard deviation σ_d is set to 3 in our experiments.

To fit the energy *minimization* framework, our data term E_d is finally written as

$$E_d(L_t) = \sum_{\mathbf{x}_t \in I_t} 1 - P(\mathbf{x}_t, L_t(\mathbf{x}_t)). \quad (7)$$

3.3. Adaptive Frame Selection with Labeling Prior

The date term in Eq. (2) involves variables $\phi_v(\cdot)$ and $\phi_o(\cdot)$, and the prior $p_o(\cdot)$. They are defined with a novel frame-selection scheme based on an observation. That is, rather than summing the matching cost over all frames, a better strategy for multiview geometry enforcement is to only *pick* frames where corresponding pixels exist (or are visible).

We introduce an effective method to search for frames that contain non-occluded matching pixels for each reference pixel $\mathbf{x}_t$. Given the initial label maps or their estimates from the previous iteration, we use the 3D warping technique [10] to warp $L_{t'}$ to the reference frame t. One example is shown in Figure 3. The label map warped from frame t' to t is denoted as $L_{t',t}$, as shown in (c). If a pixel $\mathbf{x}_t$ does not receive any label projection from frame t', the value of $L_{t',t}(\mathbf{x}_t)$ is regarded as missing, which implies that the corresponding pixel of $\mathbf{x}_t$ in frame t' is occluded.

We use this criterion to select *visible* and *invisible* frames for each pixel and denote by $\phi_v(\mathbf{x}_t)$ and $\phi_o(\mathbf{x}_t)$ respectively the set of frames where correspondences of $\mathbf{x}_t$ are visible and are occluded. Practically, we at most collect N_1 frames for $\phi_v(\mathbf{x}_t)$. N_1 is set to $16 \sim 20$ in our experiments. If the total number of frames in $\phi_v(\mathbf{x}_t)$ cannot even reach a lower limit N_2, which is generally set to 5, we add a few neighboring frames to $\phi_o(\mathbf{x}_t)$ so that $|\phi_o(\mathbf{x}_t)| + |\phi_v(\mathbf{x}_t)| = N_2$.

Note that occluded pixels have no matching costs. So if a pixel is occluded in all neighboring frames, its true disparity cannot be inferred directly. Why do we still collect frames to form $\phi_o(\mathbf{x}_t)$? It is because we found although accurate inter-frame matching is not achievable, there is a simple means to coarsely infer the disparities and object labels even in the extreme no-visible-correspondence situation using *disparity neighbors*.

Based on the fact that occluded pixels generally have small disparity values, we apply an easy but effective algorithm for *label map inpainting*. For each missing pixel $\mathbf{x}$ in the projected $L_{t',t}$, we search horizontally and vertically for four nearest neighbors that receive labels, and select the one, denoted as $\mathbf{x}^*$, with the minimum label index, as shown in Figure 3(d). The confidence to set $L_{t',t}(\mathbf{x}) = L_{t',t}(\mathbf{x}^*)$ is dependant of the distance between $\mathbf{x}$ and $\mathbf{x}^*$, which is high when the two pixels are close. We use a spatial Gaussian falloff to model the confidence

$$w_o(\mathbf{x}) = e^{-\frac{||\mathbf{x}-\mathbf{x}^*||^2}{2\sigma_w^2}}, \quad (8)$$

where $\sigma_w = 10$ empirically.

Label map hole filling does not have very high accuracy, but works pretty well when visible correspondences are not enough in estimating a reliable data cost. The labeling prior making use of this piece of information is defined as

$$p_o(\mathbf{x}_t, l, L_{t',t}) = \lambda_o \cdot w_o(\mathbf{x}_t) \frac{\beta}{\beta + |l - L_{t',t}(\mathbf{x}_t)|}, \quad (9)$$

where λ_o is the weight, and β controls the shape of the differential cost. The formulation of p_o requires that $L_t(\mathbf{x}_t)$ is similar to $L_{t',t}(\mathbf{x}_t)$ with high confidence $w_o(\mathbf{x})$.

In [11, 22], the depth/disparity maps of neighboring views are projected to the reference for depth/disparity fusion. The accuracy of the fused depth depends on the accuracy of the projected depth maps. In comparison, we use the projected *label maps* as a prior to avail selecting visible frames and stabilizing the ill-posed likelihood estimation for occluded pixels with hole filling.

This strategy is very useful for pixels near discontinuous boundaries where occlusion commonly arises, and in the

(a) (b) (c) (d) (e)

Figure 4. Result comparison using and without using the adaptive frame selection and labeling prior. (a) The first frame of the sequence. (b-c) Two estimated label maps, using and without using the adaptive frame selection scheme and the labeling prior. (d)-(e) Close-ups of (b) and (c).

meantime does not affect depth estimation for other visible pixels. Figure 4(b) and (c) (close-ups in (d) and (e)) show two results using and without using our *adaptive frame selection* scheme and the *labeling prior*. The comparison shows that our method remarkably improves segmentation and disparity estimation along the moving head, which consistently occludes the background and was very difficult to handle conventionally.

3.4. Smoothness Term

Since our label encodes disparity and segment jointly, the spatial smoothness of these two sets of variables can be maintained by only enforcing the label index smoothness, which yields a simple form

$$E_s(L_t) = \lambda_s \sum_{\mathbf{x}_t} \sum_{\mathbf{y}_t \in N(\mathbf{x}_t)} \rho(L_t(\mathbf{x}_t), L_t(\mathbf{y}_t)), \qquad (10)$$

where $N(\mathbf{x}_t)$ is the set of neighbors of pixel $\mathbf{x}_t$, and λ_s is a smoothness weight. $\rho(\cdot)$ is a robust function defined as

$$\rho(L_t(\mathbf{x}_t), L_t(\mathbf{y}_t)) = \min\{|L_t(\mathbf{x}_t) - L_t(\mathbf{y}_t)|, \eta\},$$

where $|L_t(\mathbf{x}_t) - L_t(\mathbf{y}_t)|$ measures the distance of indices between $L_t(\mathbf{x}_t)$ and $L_t(\mathbf{y}_t)$, and η truncates very large values to preserve discontinuity. This simple smoothness form can be efficiently solved by belief propagation [6] (the complexity is linear to the number of labels), and is enough even for the challenging examples shown in the paper.

4. Solving the Objective Function

In the first place, the label maps of the whole sequence are unknown. So the energy defined in (1) cannot be directly solved. We introduce a system initialization step to separately estimate a label map for each frame by removing the geometric coherence constraint $p_v(\cdot)$. Labeling prior $p_o(\cdot)$ is also omitted, simplifying the likelihood in (2) to

$$P_{init}(\mathbf{x}_t, L_t(\mathbf{x}_t)) = \frac{1}{|\phi'(\mathbf{x}_t)|} \sum_{t' \in \phi'(\mathbf{x}_t)} p_c(\mathbf{x}_t, L_t(\mathbf{x}_t), I_t, I_{t'}),$$

where $\phi'(\mathbf{x}_t)$ contains the selected frames. Without the label maps in the beginning, we resort to the temporal selection method of Kang and Szeliski [7] to pick frames where the corresponding pixels of $\mathbf{x}_t$ are visible.

Algorithm 1 Multi-Body Plane Fitting

1. Use mean shift to produce color segments $\hat{s} = \{s_i | i = 1, 2, ..., N_s\}$ in I_t.
2. **for** each segment s_i in I_t **do**
   for $k = 1, ..., K$ **do**
   Estimate the plane parameters for s_i by minimizing (11). The output includes the parameters $[a_i^k, b_i^k, c_i^k]$ and the total cost $E'^k(a_i^k, b_i^k, c_i^k)$.
   end for
   Find the optimal plane parameters $[a_i^j, b_i^j, c_i^j]$, where $j = \arg\min_k E_t^k(a_i^k, b_i^k, c_i^k)$.
   end for
3. If $E_t'^j < E_t'$, update $d_{\mathbf{x}_t} = a_i x + b_i y + c_i$ and set $S(\mathbf{x}_t) := j$ for any pixel $\mathbf{x}_t \in s_i$.

The initial objective function is correspondingly modified to

$$E'(L; \hat{I}) = \sum_{t=1}^{n} \sum_{\mathbf{x}_t \in I_t} (1 - P_{init}(\mathbf{x}_t, L_t(\mathbf{x}_t)) + \lambda_s \sum_{\mathbf{y}_t \in N(\mathbf{x}_t)} \rho(L_t(\mathbf{x}_t), L_t(\mathbf{y}_t))). \qquad (11)$$

Since the labels of different frames are not correlated in this form, we solve for L_t for each frame t separately by loopy belief propagation (BP) [6]. One resulted label map is shown in Figure 5(b). It is however erroneous especially in textureless regions.

4.1. Multi-Body Plane Fitting

To handle textureless regions and make the following refinement easier, we also incorporate color segmentation in the initialization step. The color segments are computed by the mean-shift method [3]. Then we model each color segment s_i as a 3D plane with parameters $[a_i, b_i, c_i]$ such that $d_{\mathbf{x}_t} = a_i x + b_i y + c_i$ for each pixel $\mathbf{x}_t = [x, y] \in s_i$.

With the new configuration that the scene contains multiple moving objects, traditional plane fitting methods (e.g., [20]) cannot be used. Here, we introduce a multi-body algorithm, sketched in Algorithm 1. For each color segment

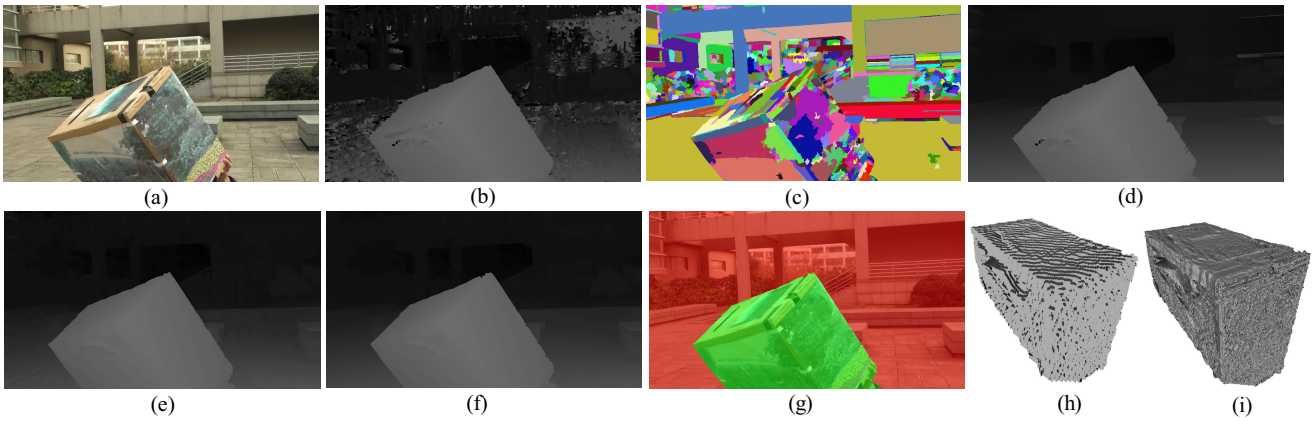

Figure 5. Intermediate results. (a) One frame from a sequence. (b) Initial label estimate without plane fitting. (c) The obtained color segments by Mean Shift method [3]. (d) The label map after plane fitting. (e) The refined label map after the first-pass optimization. (f) The refined label map after the two-pass optimization. (g) The box and background segments. (h) The reconstructed 3D surface of the box without disparity level expansion. (i) The final 3D surface after disparity level expansion.

s_i, we first assign it to the 1st object. So the camera parameters are set to $\mathbf{C}_t^1 = \{\mathbf{K}_t, \mathbf{R}_t^1, \mathbf{T}_t^1\}$. By taking all pixels in s_i into Eq. (11) while fixing the labels in all other color segments, we compute the best plane parameters $[a_i^1, b_i^1, c_i^1]$ using the method of [27]. The correspondingly minimized total cost in s_i is denoted as $E'^k(a_i^1, b_i^1, c_i^1)$.

Afterwards, we assign s_i to object 2, and repeat the above process to compute $E'^k(a_i^2, b_i^2, c_i^2)$. It continues until $[a_i^K, b_i^K, c_i^K]$ are estimated. With the K sets of possible plane parameters, s_i suits best the object with the minimum total energy, that is, $j = \arg\min_k E_t^k(a_i^k, b_i^k, c_i^k)$.

Note that assigning j to fit a plane does not necessarily yield a better result than the initial label map. We thus compare $E_t'^j$ with the initially computed cost E_t' expressed in (11) for all pixels in s_i. $E_t'^j < E_t'$ means plane fitting yields a lower-energy configuration. So the pixels in s_i need to be updated to $d_{\mathbf{x}_t} = a_i x + b_i y + c_i$ and $S(\mathbf{x}_t) = j$. On the contrary, if $E_t'^j > E_t'$, it is very likely that the segment spans multiple layers or is simply inappropriate to model the surface by a 3D plane. We do not risk updating labels in this case. Figure 5(d) demonstrates the effectiveness of this step. The initially erroneous estimates are dramatically improved, especially in textureless regions.

4.2. Iterative Spatio-Temporal Optimization

Although plane fitting is useful for frame-wise depth estimation and segmentation, due to the lack of explicit temporal coherence constraint, the independently estimated labels are not consistent, as illustrated in Figure 5(d) and our supplementary video [1]. The initial labels are occasionally wrong in some frames, which can be corrected in multiple frames in an outlier-rejection fashion making use of the geometry coherence term $p_v(\cdot)$ in Eq. (5).

Considering a pixel $\mathbf{x}$ in frame t and denoting its corresponding pixel as $\mathbf{x}'$ in frame t', if both labels $L_{t'}(\mathbf{x}')$ and $L_t(\mathbf{x})$ are correct and satisfy the color coherence constraint, $p_v(\mathbf{x}_t, l, L_{t'})$ in (5) and $p_c(\mathbf{x}_t, l, I_t, I_{t'})$ in (3) will output large values. In contrast, outliers generally cannot satisfy all constraints simultaneously, yielding very small $p_c(\cdot)p_v(\cdot)$ in the likelihood (2).

Based on the analysis, we use all terms in the data function (i.e. Eq. (7)) and progressively update the estimates by minimizing the energy (1). We process the frames sequentially starting from the first one. In optimizing label map L_t, we fix the estimates in other frames, which makes Eq. (1) be expressed as

$$E_t(L_t) = E_d(L_t) + E_s(L_t). \tag{12}$$

It is minimized by belief propagation. While processing one frame in the middle or at the back of the sequence, due to the refined labels in all frames before it, $p_v(\cdot)$ can be very reliable since it utilizes updated information. We adopt two passes of optimization to let all frames be processed with nearly even neighborhood information.

Figure 5(e) and (f) show the label maps after the first- and second-pass optimization. The first-pass optimization already corrects most of the problematic estimates. Our supplementary video can better demonstrate the temporal consistency. The obtained labels are finally decomposed into disparities and object segment indices.

Due to the use of discrete optimization, the disparities are with limited levels, as demonstrated in Figure 5(h). We densify them by a hierarchical belief propagation method [25]. In this process, the computed object segments are fixed. Figure 5(i) shows the reconstructed mesh after disparity level expansion.

[1] The supplementary video can be downloaded from the corresponding project website under http://www.cad.zju.edu.cn/home/gfzhang/

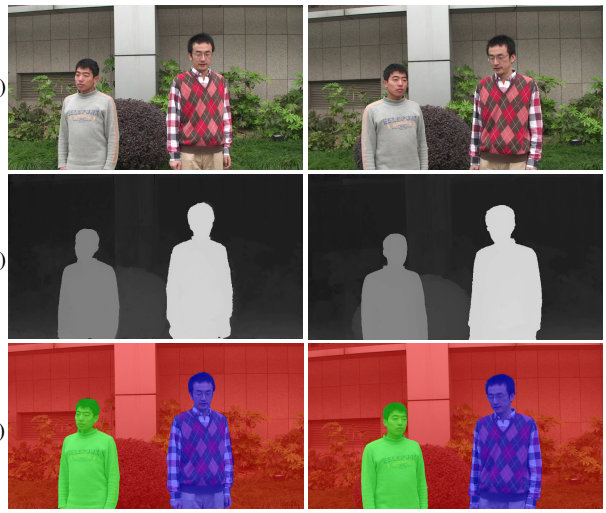

Figure 6. Three-body sequence. (a) Two selected frames. (b) The estimated label maps. (c) The estimated object masks.

5. Experimental Results

We took a few video clips by a handheld consumer digital camera. The frame resolution is 960×540 (pixels). Most of the parameters in our system are fixed. Specifically, $\lambda_s = 5/|\mathcal{L}|$, $\eta = 0.03|\mathcal{L}|$, $\lambda_o = 0.3$, $\sigma_c = 10$, $\sigma_d = 2$, and $\beta = 0.02|\mathcal{L}|$. The number of the disparity levels m_k for each object is generally set to $51 \sim 101$. Given 243 labels, our system takes about 10 minutes to process one frame (including initialization and the two-pass optimization) on a desktop computer with a 4-core Intel Xeon 2.66 GHz CPU.

Figure 6 shows a three-body example containing two persons turning around. The full sequence is included in the video. It is very challenging for accurate depth estimation and motion segmentation because occlusion arises very often and there exist large textureless regions. Our computed label maps are shown in (b), which are accurate even along boundaries. Figure 6(c) shows our high-quality object segments.

Another "Boxes" example is shown in Figure 7. The front box occludes the background and another moving box, making occlusion complex. Our method can faithfully estimate the respective depth maps and produce accurate segmentation. The example in Figure 8 contains three toy cars moving on the ground. Their depth and object segments are computed. Figure 9 demonstrates a moving car example. Strong reflection of the car surface can be noticed. The cast shadow on the road brings additional difficulties. Even with these challenges, our results are still visually compelling, except for some regions that violate the color constancy constraint in multi-view geometry – for example, the window and specular reflection surface. The extracted car has an accurate boundary.

Figure 7. "Boxes" example. (a) One frame from the input sequence. (b) The estimated label map.

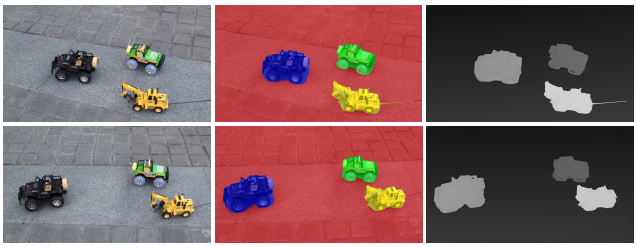

Figure 8. "Toy" example. (a) Two selected frames. (b) The estimated object mask images. (c) The estimated label maps.

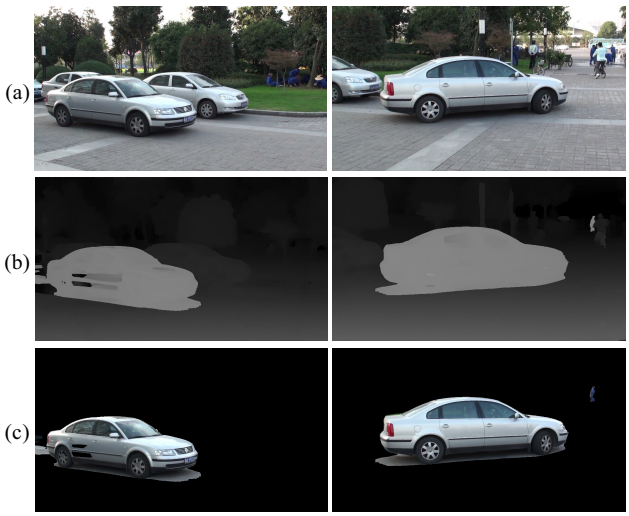

Figure 9. Challenging "Car" example. (a) Two frames from the input sequence. (b) The estimated label maps. (c) The extracted car images.

6. Conclusions

In this paper, we have presented a novel multi-body stereo method for constructing high-quality depth maps and for segmentation of several moving rigid objects from an input monocular image sequence. The new multi-body stereo label representation couples depth and segmentation indices, making it possible to employ optimization to simultaneously compute these two sets of variables. A multi-body plane fitting method is introduced to improve initial estimates in textureless regions, together with disparity hole filling to offer additional matching information for occluded

pixels.

Currently, our method can only handle independently moving rigid objects. Nonrigid objects in this system will still be classified as rigid ones. Handling them properly will be our future work.

Acknowledgements

This work is supported by the 973 program of China (No. 2009CB320802), NSF of China (No. 60903135), China Postdoctoral Science Foundation funded project (No. 20100470092), the Research Grants Council of the Hong Kong Special Administrative Region (under General Research Fund – project No. 412911), and by a research grant from Microsoft Research Asia through the joint lab with Zhejiang University.

References

[1] S. Ayer and H. S. Sawhney. Layered representation of motion video using robust maximum-likelihood estimation of mixture models and mdl encoding. In *ICCV*, pages 777–784, 1995. 1

[2] A. Buchanan and A. W. Fitzgibbon. Interactive feature tracking using k-d trees and dynamic programming. In *CVPR (1)*, pages 626–633, 2006. 2

[3] D. Comaniciu, P. Meer, and S. Member. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:603–619, 2002. 2, 5, 6

[4] J. P. Costeira and T. Kanade. A multi-body factorization method for motion analysis. In *ICCV*, pages 1071–, 1995. 1

[5] A. Criminisi, G. Cross, A. Blake, and V. Kolmogorov. Bilayer segmentation of live video. In *CVPR (1)*, pages 53–60, 2006. 2

[6] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient belief propagation for early vision. *International Journal of Computer Vision*, 70(1):41–54, 2006. 5

[7] S. B. Kang and R. Szeliski. Extracting view-dependent depth maps from a collection of images. *International Journal of Computer Vision*, 58(2):139–163, 2004. 5

[8] M. P. Kumar, P. H. S. Torr, and A. Zisserman. Learning layered motion segmentations of video. *International Journal of Computer Vision*, 76(3):301–319, 2008. 1

[9] L. Ladicky, P. Sturgess, C. Russell, S. Sengupta, Y. Bastanlar, W. Clocksin, and P. H. S. Torr. Joint optimisation for object class segmentation and dense stereo reconstruction. In *BMVC*, pages 1–11, 2010. 1

[10] W. R. Mark, L. McMillan, and G. Bishop. Post-rendering 3D warping. In *SI3D*, pages 7–16, 180, 1997. 4

[11] P. Merrell, A. Akbarzadeh, L. Wang, P. Mordohai, J.-M. Frahm, R. Yang, D. Nistér, and M. Pollefeys. Real-time visibility-based fusion of depth maps. In *ICCV*, 2007. 4

[12] K. E. Ozden, K. Schindler, and L. J. V. Gool. Simultaneous segmentation and 3D reconstruction of monocular image sequences. In *ICCV*, pages 1–8, 2007. 1, 2

[13] S. Rao, R. Tron, R. Vidal, and Y. Ma. Motion segmentation in the presence of outlying, incomplete, or corrupted trajectories. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(10):1832–1845, 2010. 1

[14] F. Rothganger, S. Lazebnik, C. Schmid, and J. Ponce. Segmenting, modeling, and matching video clips containing multiple moving objects. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(3):477–491, 2007. 1

[15] K. Schindler, J. U, and H. Wang. Perspective -view multibody structure-and-motion through model selection. In *ECCV (1)*, pages 606–619, 2006. 1

[16] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *CVPR (1)*, pages 519–528, 2006. 1

[17] E. Sharon, M. Galun, D. Sharon, R. Basri, and A. Brandt. Hierarchy and adaptivity in segmenting visual scenes. *Nature*, 442(7104):719–846, June 2006. 2

[18] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:888–905, 2000. 2

[19] J. Sun, W. Zhang, X. Tang, and H.-Y. Shum. Background cut. In *ECCV (2)*, pages 628–641, 2006. 2

[20] H. Tao, H. S. Sawhney, and R. Kumar. A global matching framework for stereo computation. In *ICCV*, pages 532–539, 2001. 5

[21] R. Tron and R. Vidal. A benchmark for the comparison of 3-D motion segmentation algorithms. In *CVPR*, 2007. 1

[22] C. Unger, E. Wahl, P. Sturm, and S. Ilic. Probabilistic disparity fusion for real-time motion-stereo. In *ACCV*, 2010. 4

[23] Y. Weiss and E. H. Adelson. A unified mixture framework for motion segmentation: Incorporating spatial coherence and estimating the number of models. In *CVPR*, pages 321–326, 1996. 1

[24] C. Zhang, L. Wang, and R. Yang. Semantic segmentation of urban scenes using dense depth maps. In *ECCV (4)*, pages 708–721, 2010. 1

[25] G. Zhang, Z. Dong, J. Jia, L. Wan, T.-T. Wong, and H. Bao. Refilming with depth-inferred videos. *IEEE Trans. Vis. Comput. Graph.*, 15(5):828–840, 2009. 6

[26] G. Zhang, J. Jia, W. Hua, and H. Bao. Robust bilayer segmentation and motion/depth estimation with a handheld camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(3):603–617, 2011. 2

[27] G. Zhang, J. Jia, T.-T. Wong, and H. Bao. Consistent depth maps recovery from a video sequence. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(6):974–988, 2009. 3, 4, 6

[28] G. Zhang, X. Qin, W. Hua, T.-T. Wong, P.-A. Heng, and H. Bao. Robust metric reconstruction from challenging video sequences. In *CVPR*, 2007. 2

[29] C. L. Zitnick, N. Jojic, and S. B. Kang. Consistent segmentation for optical flow estimation. In *ICCV*, volume 2, pages 1308–1315, 2005. 1