A General Inlier Estimation for Moving Camera Motion Segmentation

XUEFENG LIANG^{1,a)} CUICUI ZHANG^{1,b)} TAKASHI MATSUYAMA^{1,c)}

Received: January 29, 2015, Accepted: September 14, 2015, Released: December 16, 2015

Abstract: In moving camera videos, motion segmentation is often achieved by determining the motion coherence of each moving object. However, it is a nontrivial task on optical flow due to two problems: 1) Optical flow of the camera motions in 3D world consists of three primary 2D motion flows: translation, rotation, and radial flow. Their coherence analysis is done by a variety of models, and further requires plenty of priors in existing frameworks; 2) A moving camera introduces 3D motion, the depth discontinuities cause the motion discontinuities that severely break down the coherence. Meanwhile, the mixture of the camera motion and moving objects' motions make it difficult to clearly identify foreground and background. In this work, our solution is to transform the optical flow into a potential space where the coherence of the background flow field is easily modeled by a low order polynomial. To this end, we first amend the Helmholtz-Hodge Decomposition by adding coherence constraints, which can transform translation, rotation, and radial flow fields to two potential surfaces under a unified framework. Secondly, we introduce an Incoherence Map and a progressive Quad-Tree partition to reject moving objects and motion discontinuities. Finally, the low order polynomial is achieved from the rest flow samples on two potentials. We present results on more than twenty videos from four benchmarks. Extensive experiments demonstrate better performance in dealing with challenging scenes with complex backgrounds. Our method improves the segmentation accuracy of state-of-the-arts by 10%~30%.

Keywords: Helmholtz-Hodge Decomposition, incoherence map, progressive Quad-Tree, moving camera video, motion segmentation, inlier

1. Introduction

The rapid increase of mobile cameras (handhold camera, wearable camera, etc.) creates more challenges for video analysis. One of them is that appearance motions are no longer simple in such scenes where multiple objects could move independently with respect to the camera motion. It is named 3D motion. Common schemes in 3D motion segmentation use either optical flow or trajectories as a cue. As optical flow is a dense projection of motion from 3D world to 2D image plane, it can be directly used for clustering or to compensate for the camera motion. Therefore, the pixel-wise model is often used for segmentation [1], [2], [3], [4].

However, here are two major drawbacks of using optical flow. 1) Camera motions in 3D world cause three primary motion flows in optical flow: *translation*, *rotation*, and *radial flow*. The well accepted interpretation of the flow vector is based on Cartesian coordinate system with two bases x , y . Then, a motion vector in 2D is denoted by u , v . For a translation, the direction and speed can be interpreted as invariants $\arctan(u/v)$ and $\sqrt{u^2 + v^2}$ respectively, which depend on the scene depth and camera motion only. But for a certain rotation or radial flow, both $\arctan(u/v)$ and $\sqrt{u^2 + v^2}$ change with x , y changing on image plan. Then,

no invariant is for their interpretation under the Cartesian coordinate system. For example, Fig. 1 (b) and Fig. 4 (b) involve the radial flow and rotation, respectively. The direction and magnitude of flow vectors are changing w.r.t the change of x , y . To our best knowledge, no scheme could well handle all these three flows based on dense flow field. 2) Optical flow of objects depends on the distance to the camera in 3D world. Varied depths cause different magnitudes of the flow. This may make a clustering algorithm to label backgrounds at different depths as separate objects although they are static in the real world. In Fig. 1 (b), there is a significant motion discontinuity between the stopped car and the far away background. To handle motion discontinuities, other works need auxiliary information (e.g., color, edge energy) to merge small segments into one by a post-processing.

As the flow field of the static background (inlier) in optical flow is caused by the camera motion, it should be globally coherent. But, existing optical flow algorithms produce many errors at the depth discontinuity, and make those flows incoherent with the camera motion. For a moving object in 3D scene, its flow field (outlier), however, is caused by not only its own motion but also the camera motion. So, an outlier consists of both incoherent and coherent flows w.r.t. the camera motion. Thus, the coherence in optical flow can be a cue for 3D motion segmentation. In this work, our objective is to find an appropriate polynomial for inlier modeling according to its coherence, which requires little prior knowledge and post-processing. To this end, the first challenge is how to put three primary motion flows into one scheme. Helmholtz-Hodge decomposition (HHD) was initially developed

¹ IST, Graduate School of Informatics, Kyoto University, Kyoto 606-8501, Japan

^{a)} xliang@i.kyoto-u.ac.jp

^{b)} zhang@vision.kuee.kyoto-u.ac.jp

^{c)} tm@i.kyoto-u.ac.jp

Fig. 1 (a) A traffic scene where two kids are running from left to right. A car stops at the left side. The camera is moving towards the stop line. (b) The original optical flow. Besides two outliers, here exists a significant flow discontinuity between the left side car and far away background. (c) The potential of divergence-free component. (d) The potential of curl-free component. (e) Incoherence Map IM . (f) Progressive Quad-Tree Partition on IM to reject the outliers, motion discontinuities, and noise. (g) The estimated inlier. It has been rather coherent. (h) Result of 3D motion segmentation. It detects outliers only.

to characterize the rotation and radial flow by curl-divergence regularization. HHD can decompose an arbitrary motion field into *curl-free* and *divergence-free* components through finding their unique corresponding scalar and vector potentials regardless of whether or not it is coherent. To ensure the coherence, we amend conventional HHD by adding two constraints: piece-wise smoothness, and global minimization. Nevertheless, the obtained two potentials consist of the major coherent inlier and a little coherent outliers. To better estimate inlier, we introduce an Incoherence Map (IM) by subtracting the projection of two potentials from optical flow. It intuitively depicts the outliers and motion discontinuities. Moreover, a progressive Quad-Tree partition is proposed for precisely labeling outliers and motion discontinuities on IM , and rejecting them from two potentials. Outliers are completely excluded in inlier estimation. The motion discontinuities are also excluded, but do not affect the coherent flows at the fields with different depths. Afterwards, our objective can be achieved by approximating the low order polynomials using rest samples on two potentials. In short, our contributions are three-fold:

- Unlike other methods using a variety of models and priors, we put three 2D primary motion flows (*translation*, *rotation*, and *radial flow*) under a unified framework: the amended Helmholtz-Hodge Decomposition. Thus, primary motion flows are interpreted by two invariants: *curl-free* and *divergence-free* components which require little prior knowledge.
- To identify the inlier and outliers, we propose an Incoherence Map and a progressive Quad-Tree partition to reject moving objects and motion discontinuities from background. It also remarkably alleviates the effect of depth discontinuity in 3D motion.
- Due to the coherence introduced by camera motion, we propose a low order polynomial function to approximate the in-

lier. Then outliers are able to be segmented and recovered. These contributions are evaluated by extensive experiments. Results show that the proposed method is able to handle more challenging motion flows, and outperform the state-of-the-arts up to 10%~30% on segmentation accuracy.

Before we begin a more detailed description, it is worthwhile to highlight some related works in next section. Afterwards, we provide further technical details of inlier modeling in Section 3, and outliers detection in Section 4. In experiment section, we demonstrate the performance of the proposed method by comparing with the state-of-the-arts, and we conclude this paper in conclusion and future work section.

2. Related works

Other approaches for 3D motion segmentation can be categorized as (1) using optical flow, and (2) trajectories clustering.

2.1 Using Optical Flow

Methods using optical flow perform pixel-level segmentation on image plane motion. They model the camera-induced 2D motion flow by a parametric transformation ranging from translation to perspective transformation [5]. Pixels that are consistent with the estimated model are supposed to be inlier, while others are outliers. Nevertheless, the parametric model is an approximation of a few restricted camera motions and scene structure, and severely relies on prior-knowledge for model selection. In addition, outliers commonly violate the inlier estimation, it is ideal to remove outliers first. A regression scheme, using gradient descent (GD) [5] or least squares (LS) [6], was applied to refine the inlier model by iteratively excluding outliers. A statistical method, RANSAC [7], estimates the inlier on the data containing outliers by iteratively updating the probability of inlier. Chen and Bajic [1] proposed an outlier rejection filter that explicitly filters motion vectors by checking their similarity in a pre-defined

window. Chen and Bajic [2], and Qian and Bajic [3] proposed a joint global motion estimation, which iteratively updates the inlier model by segmenting outliers out. Although these methods have achieved great progress in dealing with the independent motions, they are very likely to over-segment objects due to the motion bias introduced by camera motion [2]. Moreover, the magnitude of optical flow is also determined by the depth. The flow discontinuities cause by depth variation are often recognized as outlier as well, even they share the same real-world motion. Recently, Narayana et al. [4] proposed a method using the direction of motion flow only. It works well for 2D translation, but has difficulty with rotation and radial flow.

2.2 Trajectories Clustering

The basic idea of trajectories clustering is to track selected local features in the scene, and then group them into different subspaces [8], [9] to achieve segmentation. Ichimura [10] introduced a factorization method to directly factor the trajectory matrix of multiple motions into sub-matrices of different motions, but requires the motion to be independent from one another. Thus, it was designed for the static camera videos. Vidal et al. [11] proposed a Generalized Principal Component Analysis (GPCA), which can partially handle dependent motions. Sugaya and Kanatani [12] developed a statistical Multi-Stage Learning (MSL), which is based on Costeira and Kanade's factorization method (CK) [13] and Kanatani's subspace separation method (SS) [14]. The weak point is that above methods assume an orthographic camera model. It is a simplified perspective camera model requiring a long focal length to avoid any perspective distortions due to varied 3D depths, therefore cannot meet many real world cases. In recent developments, Brox and Malik [15] segment trajectories by computing the pairwise distances between all trajectories and finding a low-dimensional embedding using spectral clustering. Later, Ochs and Brox [16] improved the spectral clustering by using higher order interactions that consider triplets of trajectories. Elqursh and Elgammal [17] proposed an online extension of spectral clustering by considering trajectories from multiple frames. But they required post-processing for merging. Kwak et al. [18] used a Bayesian filtering framework that combines block-based color appearance models with separate motion models for segmentation. However, they require a special initialization procedure in the first frame. Although being able to process certain complex motions, trajectories clustering methods rely much on the presence of strong features, robust feature extraction, tracking, and clustering algorithms to have trajectories as motion cues [16], [19]. In addition, as they only output a segmentation of sparse key features, post-processing is necessary for producing a dense segmentation.

In contrast, our method is a frame-to-frame scheme that requires neither trajectories from multiple frames nor special initialization and post-processing. In additional, it handles all three primary flows in one scheme, and works on a rather wide bank of videos.

Recently, Muntaz et al. [20] proposed a joint foreground-background mixture model (FBM) to address the dynamic texture problem. It divides a video into spatiotemporal overlapping

patches, and encodes the appearance of each patch at time t by location-specific dynamic texture (DT) components and global DT components. By jointly updating the local and global DT components, FBM can well estimate the background. It is robust to the scene with dynamic texture motion, but, does not show advantage on the moving camera video data.

3. Modeling Inlier of Optical Flow

3.1 Models of Optical Flow and 3D Motion Segmentation

Let X, Y, Z denote the horizontal, vertical and depth axes in a real world Cartesian coordinate, and let x, y denote the corresponding coordinates in the image plane. The image plane is located at the focal length: $Z' = f$. In 3D scene, the camera motion has two components: a translation $T = (T_X, T_Y, T_Z)$ and a rotation $R = (R_X, R_Y, R_Z)$. They are always coherent (continues and smooth in both direction and magnitude) in a short time interval Δt . Then, the resulting 2D optical flow u and v in the x and y image axes are Ref. [21].

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} -f(\frac{T_X}{Z} + R_Y) + x\frac{T_Z}{Z} + yR_Z - x^2\frac{R_Y}{f} + xy\frac{R_X}{f} \\ -f(\frac{T_Y}{Z} - R_X) + y\frac{T_Z}{Z} - xR_Z + y^2\frac{R_X}{f} - xy\frac{R_Y}{f} \end{bmatrix}. \quad (1)$$

Beside constant f , camera motion T and R , and image coordinates x and y , it is noted from Eq. (1) that u and v are functions with depth Z too. Thus, an ideal optical flow field ($\mathcal{OF}$) in 3D scene is a collection of 2D $\mathcal{OF}$ s of n static planar surfaces in background (named *inlier*, $\mathcal{F}_{in}$) and $\mathcal{OF}$ s of m foreground moving objects (named *outlier*, $\mathcal{F}_{out}$). We can formulate them as

$$\mathcal{OF} = \mathcal{F}_{in} + \mathcal{F}_{out}, \quad (2)$$

where $\mathcal{F}_{in} = [\mathcal{F}_{in}(B_1) \mathcal{F}_{in}(B_2) \dots \mathcal{F}_{in}(B_n)]$, $\mathcal{F}_{out} = [\mathcal{F}_{out}(O_1) \mathcal{F}_{out}(O_2) \dots \mathcal{F}_{out}(O_m)]$, B_i denotes a static planar background surface and O_j denotes a moving object. Please note that $\mathcal{F}_{out}(O_j)$ is incoherent with $\mathcal{F}_{in}$ because its own motion does not match the camera motion.

Theoretically speaking, each $\mathcal{F}_{in}(B_i)$ or $\mathcal{F}_{out}(O_j)$ could be approximated by a polynomial. Thus, entire $\mathcal{OF}$ could be also approximated by a high order polynomial $\mathcal{P}$.

$$\mathcal{OF} = [\mathcal{F}_{in}(B_1) \dots \mathcal{F}_{in}(B_n) \mathcal{F}_{out}(O_1) \dots \mathcal{F}_{out}(O_m)] \approx \mathcal{P}, \quad (3)$$

where high order is required due to the outliers, motion discontinuities caused by depth discontinuities, and noise.

To segment 3D motion, many studies tried to model outliers directly. Moving objects, however, can be either rigid or non-rigid, outlier modeling is a nontrivial task. Instead of complex modeling and auxiliary constraints, we model inlier by a general approach in this work other than modeling outliers. As the inlier is caused by camera motion, each $\mathcal{F}_{in}(B_i)$ must be coherent, and can be modeled by a simple polynomial $\mathcal{P}_i$. But Eq. (1) shows $\mathcal{F}_{in}(B_i)$ is a function with Z in translation and radial flow fields. Modeling $\mathcal{F}_{in}$ still faces difficulties.

Most optical flow algorithms share a common assumption of local motion smoothness, and apply an optimization to minimize the global error. The difference among them only focuses on implementations of the optimization. This strategy is ideally designed for the motion of one planar surface. But, it is also applied

on the object's boundaries, where depths vary, because algorithms do not know where the depth discontinuities are. For inlier $\mathcal{F}_{in}$, at the place where depth variation is not significant, optical flow algorithms give rather smooth flow field and connect motion fields of two adjacent planar surfaces. By the same token, algorithms often produce many errors at places having significant depth discontinuity. These incorrect flows are incoherent with camera motion, but are minority in $\mathcal{F}_{in}$. Then, $\mathcal{F}_{in}$ can be reformulated as

$$\mathcal{F}_{in} = a_1 \mathcal{F}_{in}^{coherent} + b_1 \mathcal{F}_{in}^{incoherent}, \quad a_1 \gg b_1, \quad (4)$$

where a and b are quantity coefficients.

The primary reason of the moving objects standing out in a moving camera video is that their motions are incoherent with camera motion and result in significant variances on optical flow field $\mathcal{OF}$. Besides, $\mathcal{F}_{out}$ is partially caused by camera motion as well, which is coherent with $\mathcal{F}_{in}$, but is minority. Now, $\mathcal{F}_{out}$ can be reformulated as

$$\mathcal{F}_{out} = a_2 \mathcal{F}_{out}^{coherent} + b_2 \mathcal{F}_{out}^{incoherent}, \quad b_2 \gg a_2. \quad (5)$$

One major difficulty in 3D motion segmentation is induced by the mixed flow field of camera motion and moving objects' motions. These dependent motions lead to indistinctness of the difference between inlier $\mathcal{F}_{in}$ and outliers $\mathcal{F}_{out}$. To segment outliers from inlier, our objective, therefore, becomes finding an appropriate polynomial

$$\mathcal{P}' \approx a_1 \mathcal{F}_{in}^{coherent} + a_2 \mathcal{F}_{out}^{coherent}, \quad (6)$$

which rejects incoherence ($b_1 \mathcal{F}_{in}^{incoherent} + b_2 \mathcal{F}_{out}^{incoherent}$) in inlier and outliers.

3.2 Three Primary Motion Flows and Their Potentials

Equation (1) shows motion vectors in $\mathcal{OF}$ caused by camera motion can be decomposed into two components: V_T and V_R representing camera translation and rotation, respectively.

$$V_T = \left[\frac{xT_Z - fT_X}{Z}, \frac{yT_Z - fT_Y}{Z} \right]^T, \\ V_R = \begin{bmatrix} yR_Z - x^2 \frac{R_Y}{f} + xy \frac{R_X}{f} - fR_Y \\ -xR_Z + y^2 \frac{R_X}{f} - xy \frac{R_Y}{f} + fR_X \end{bmatrix}.$$

To simplify analysis, motion vectors caused by camera motion can be further decomposed into three primary components:

- (1) V_{T_x} caused by camera translation on image plan $\mathbb{X}(x, y)$:

$$V_{T_x} = \left[-\frac{fT_X}{Z}, -\frac{fT_Y}{Z} \right], \quad \begin{cases} \arctan(V_{T_x}) = \frac{T_X}{T_Y}, \\ |V_{T_x}| = \frac{f}{Z} \sqrt{T_X^2 + T_Y^2}. \end{cases} \quad (7)$$

It is noted that the direction of flow is independent on the depth Z , and the magnitude is inversely proportional to the depth Z . Equation (7) indicates that if the depth variation is not significant in 3D scene (e.g., the static background is rather far from the camera), the coherence of both direction and magnitude is preserved. If background is pretty close to the camera, the depth discontinuity will lead to motion

discontinuity that is incoherent with camera motion.

- (2) V_{T_z} caused by camera translation along Z axis. It presents a radial flow field with the origin at the focus-of-expansion.

$$V_{T_z} = \left[\frac{xT_Z}{Z}, \frac{yT_Z}{Z} \right], \quad \begin{cases} \arctan(V_{T_z}) = \frac{x}{y}, \\ |V_{T_z}| = \frac{T_Z}{Z} \sqrt{x^2 + y^2}. \end{cases} \quad (8)$$

It is noted that the flow direction is dependent on neither the depth Z nor camera motion, but only determined by image plan coordinates x and y . The magnitude is also inversely proportional to the depth Z . Analogically, the coherent flow field is maintained at the place having less depth variation. But, the incoherence occurs at the place having significant depth discontinuity. Please see Fig. 1 (b) as an example.

- (3) V_R caused by camera rotation along an axis parallel with the camera optical axis Z . Please note that camera rotation, which rotates along image axes x or y in a short time interval Δt , could be simulated as a translation. In this work, we consider camera rotation perpendicular to the image plan only.

$$V_R = [yR_Z, -xR_Z], \quad \begin{cases} \arctan(V_R) = -\frac{y}{x}, \\ |V_R| = R_Z \sqrt{x^2 + y^2}. \end{cases} \quad (9)$$

It is noted that both direction and magnitude of V_R are independent on depth Z , but dependent on image plan coordinates x and y . This indicates that a 3D motion segmentation can reduce to a 2D motion segmentation while camera solely rotates perpendicular to the image plan. Obviously, the coherence of flow field is preserved.

Consequently, an arbitrary optical flow field $\mathcal{OF}$ can be represented by a combination of above three primary flows as:

$$\mathcal{OF} = \alpha V_{T_x} + \beta V_{T_z} + \gamma V_R,$$

where α, β and γ are quantity coefficients.

Analogically, Prof. Helmholtz explained that the motion of a volume element in 3D space consists of: 1) *expansion or contraction*, 2) *rotation*, and 3) *translation*. The expansion/contraction (radial flow field) can be represented as the gradient of a scalar potential function because it is irrotational. The rotation can be represented as the curl of a vector potential function since it is incompressible. Translation however, being neither compressible nor rotational can be represented as either the gradient of a scalar potential, or the curl of a vector potential [22]. It is named by *Helmholtz-Hodge Decomposition* (HHD) [23]. According to HHD, any flow field in our work consists of two components:

- (1) **Curl-free component** representing divergence (radial flow) and translation because they are irrotational.

$$\theta = \nabla \cdot \mathcal{OF} = V_{T_z} + V_{T_x}$$

- (2) **Divergence-free component** representing curl (rotation) and translation because they are incompressible.

$$\vec{\omega} = \nabla \times \mathcal{OF} = V_{T_R} + V_{T_x}$$

Go a step further, curl-free and divergence-free components

can be expressed as the curl of a vector potential $\vec{W}$ and the gradient of a scalar potential E ,

$$\mathcal{OF} = \nabla E + \nabla \times \vec{W}. \quad (10)$$

where

$$E(x) = -\frac{1}{4\pi} \int \frac{\theta(x')}{|x - x'|} dx', \quad \vec{W}(x) = -\frac{1}{4\pi} \int \frac{\vec{\omega}(x')}{|x - x'|} dx',$$

and $x \in \mathbb{R}^3$. Actually, θ and $\vec{\omega}$ are what we expect. In reality, E and $\vec{W}$ are computed by energy minimization,

$$\min F(E) = \min \int (\mathcal{OF} - \nabla E)^2,$$

$$\min G(\vec{W}) = \min \int (\mathcal{OF} - \nabla \times \vec{W})^2.$$

Theoretically speaking, HHD can decompose an arbitrary motion flow field into curl-free component ∇E and divergence-free component $\nabla \times \vec{W}$, regardless of whether or not it is coherent. But, we expect coherent potentials for approximating the coherent flow field caused by camera motion, see Eq. (6). To ensure the coherence, our work amended the conventional HHD by adding the first constraint: 1) **piece-wise smooth $\mathcal{S}$** . Meanwhile, we make an assumption $\mathcal{F}_{in} > \mathcal{F}_{out}$, and add the second constraint: 2) **global minimization** at entire motion flow field Ω , which ensures the optimization to minimize the inlier, other than the outliers.

$$\begin{aligned} \arg \min_{\mathcal{S}} F(E) &= \arg \min_{\mathcal{S}} \int_{\Omega} (\mathcal{OF} - \nabla E)^2 d\Omega, \\ \arg \min_{\mathcal{S}} G(\vec{W}) &= \arg \min_{\mathcal{S}} \int_{\Omega} (\mathcal{OF} - \nabla \times \vec{W})^2 d\Omega. \end{aligned} \quad (11)$$

Please see Fig. 1 (c) and (d), they are the divergence-free and curl-free potentials respectively. Therefore, these two potential E and $\vec{W}$ could be our objective: coherent surfaces which can be formulated by low order polynomial $\mathcal{P}'$.

3.3 Incoherence Map and Incoherence Labeling

Due to the global optimization and piece-wise smooth constraint, our amended implementation keeps most coherent flow (95%~99%) into two potentials, but definitely rejects incoherent flow $\mathcal{F}_{in}^{incoherent}$ and $\mathcal{F}_{out}^{incoherent}$. Then, the majority of outliers, motion discontinuities and noise, which are not decomposed into ∇E and $\nabla \times \vec{W}$, will rest in a remainder. Please note that two potentials are not able to be directly used for the inlier estimation, as they still contain a small quantity of $\mathcal{F}_{out}^{coherent}$. Thus, we use this remainder to draw an **Incoherence Map (IM)** to label incoherent flows in $\mathcal{OF}$.

First, we estimate the coherent flow field presented in the curl-free and divergence-free components by a linear combination as:

$$\mathcal{V} = \alpha(\nabla E) + \beta(\nabla \times \vec{W}), \quad (12)$$

where $\mathcal{V} \subseteq \mathcal{OF}$, α and β are quantity coefficients that indicate how much ∇E and $\nabla \times \vec{W}$ are involved in $\mathcal{OF}$. We use a distance to determine α and β .

$$d_{\theta} = \int_{\Omega} \frac{|\mathcal{OF} - \nabla E|}{|\mathcal{OF}|} d\Omega, \quad d_{\omega} = \int_{\Omega} \frac{|\mathcal{OF} - \nabla \times \vec{W}|}{|\mathcal{OF}|} d\Omega, \quad (13)$$

where d_{θ} represents the distance between the curl-free component and the optical flow field, and d_{ω} represents the distance between the divergence-free component and the optical flow field. Then, α and β are determined as follows:

- While $d_{\theta} < 0.5$ and $d_{\omega} > 0.5$, it implies the optical flow looks more like a radial flow field (curl-free component).
- While $d_{\theta} > 0.5$ and $d_{\omega} < 0.5$, it implies the optical flow looks more like a rotation field (divergence-free component).

Under above two cases,

$$\alpha = \frac{d_{\theta}}{d_{\theta} + d_{\omega}}, \quad \beta = \frac{d_{\omega}}{d_{\theta} + d_{\omega}}.$$

- While $d_{\theta} < 0.5$ and $d_{\omega} < 0.5$, it means both components are similar to the input optical flow, and implies a translation. Then,

$$\alpha = 0.5, \quad \beta = 0.5.$$

Afterwards, we can draw the Incoherence Map from remainder by

$$IM = \mathcal{OF} - \alpha(\nabla E) - \beta(\nabla \times \vec{W}). \quad (14)$$

Please see Fig. 1 (e), it clearly reveals the outliers, depth discontinuities and computation error.

IM makes outliers and motion discontinuities labeling much easier. However, Eq. (12) and Eq. (14) show that IM consists of a small portion of inlier as well. The direction and magnitude of this inlier varies at different location in IM. So, a simple threshold is not possible to label all outliers. To ensure accurate labeling, we introduce a **Progressive Quad-Tree Partition** on IM, which considers the both local and global variations. The basic idea is that it partition IM into quadrants recursively if a quadrant is not coherent. The partition is called by the following two conditions: (1) the variance of a sub-quadrant $var(\Omega_i)$ is larger than $t * var(\Omega)$;

(2) the mean of a sub-quadrant $mean(\Omega_i)$ is larger than $mean(\Omega)$. where var and $mean$ calculate the variance and mean of flow direction and magnitude, respectively. t is a threshold. Ω is the entire IM. The condition 1 is for detecting the motion discontinuities including the boundaries of outliers and noise, where the flow value changes violently. The condition 2 is for detecting outliers' body, where the outliers' flow differs from inlier's flow because inlier has been almost canceled by two potentials in IM. Partition performs until no region can be split further. The smallest regions represent outliers, motion discontinuities and noise. The rest larger regions represent the coherent inlier, and will be involved in inlier approximation in next section.

The threshold t determines how the Quad-Tree partitions IM. Since local deformations usually vary on different IMs, it is rather difficult to find the best partition using one threshold. We, therefore, define a set of thresholds in a descending order, and introduce a progressive Quad-Tree partition. The t is initially set to 1, and reduces with a step 0.05 for next partition. The procedure stops when the difference between the current Quad-Tree QT' and the previous one QT is less than a convergence value ε . The updated QT' is used for labeling incoherence in IM. The pseudo-code is given below.

Fig. 2 (a) The outliers map from Eq. (17). The flow discontinuities have become very weak. (b) The potential of outliers. (c) The labeled outliers.

Data: IM
Result: Labeled incoherence
 $t = 1; QT = 0;$
 Split(Ω);
while $|QT' - QT| > \varepsilon$ **do**
 for each quadrant Ω_i **do**
 if $\text{var}(\Omega_i) > t * \text{var}(\Omega) \parallel \text{mean}(\Omega_i) > \text{mean}(\Omega)$ **then**
 Split(Ω_i);
 Go to for loop;
 end
 end
 Update QT' and QT ;
 $t = t - 0.05$;
end
 Mark all smallest blocks as incoherence;

Algorithm 1: Progressive Quad-Tree Partition

Figure 1 (f) shows that the progressive Quad-Tree partition effectively labels outliers and depth discontinuities on IM.

3.4 Inlier Estimation

Multi-parametric models had been conducted to recover the inlier [5]. They were designed for camera motions ranging from simple translation to complex perspective transformation. But, the limitation is that a prior knowledge of motion structure is required to select an appropriate model. Nevertheless, this prior knowledge is not always available in real data. By contrast, we employ a general solution, polynomial surface fitting, using d -order polynomial

$$\mathcal{P} = a_{d0}x^d + a_{0d}y^d + \cdots + a_{ij}x^i y^j + \cdots + a_{10}x + a_{01}y + a_{00}, \quad (15)$$

to estimate inlier $\mathcal{F}_{in}$ from two potentials E and $\vec{W}$. The advantage is that it requires little prior knowledge.

It is known that high order terms in Eq. (15) present the high frequency signals (incoherence: outliers, motion discontinuities and noise in this work). Thanks to IM and algorithm 1, the incoherence has been labeled and rejected in the subsequent process. As explained in Eq. (6) and Eq. (11), our objective is to find low order polynomial $\mathcal{P}'$ which expresses the coherent inlier. The inlier estimation, eventually, can be performed by sampling the rest of the flows on two potentials E and $\vec{W}$. Since outliers and noise are completely excluded, surface fitting utilizes nearby samples to approximate the inlier. Thus, the result is rather coherent with the camera motion. For motion discontinuities, surface fitting interpolates the samples at both sides of discontinuity to approximate the violent flow change. So, the result also presents the trend of rapid flow change. But the gradient of the change becomes less than the one on original $\mathcal{OF}$. In our work, a polynomial of $d = 5$

is employed to produce coherent and accurate potentials E' and $\vec{W}'$ using the samples after rejecting incoherence. The final inlier is estimated by a linear combination where α, β have been determined in Section 2.3,

$$\mathcal{F}'_{in} = \alpha \nabla E' + \beta \nabla \times \vec{W}'. \quad (16)$$

Figure 1 (g) shows the estimated inlier. It demonstrates that our method approximates the inlier rather coherent, and the outliers have been excluded effectively.

4. Outliers Detection

By using an estimated inlier, outliers $\mathcal{F}_{out}$ can be directly detected by subtracting the $\mathcal{F}'_{in}$ from the original optical flow $\mathcal{OF}$,

$$\mathcal{F}_{out} \cong \mathcal{OF} - \mathcal{F}'_{in}. \quad (17)$$

Please see Fig. 2 (a). However, low order polynomial surface fitting has a defect that it better fits the data with dense samples but goes wild at the edges of the original domain Ω due to lack of adequate samples. To reduce the error, the result is filtered by the mean-curvature of the original potential. The final segmentation is obtained subsequently by assigning binary labels on the true outliers. We will use the segmentation result to evaluate the performance of proposed method in experiments. Figure 2 (c) and (b) show the detected outliers and their potentials, respectively. Figure 1 (h) shows the 3D segmentation result.

5. Experiments

The performance of proposed method is evaluated on four benchmark datasets: Hopkins [24], Berkeley Motion Segmentation [15], Complex Background [4], SegTrack [25], and FBDynScn [20]. The Hopkins dataset contains video sequences along with the features extracted and tracked in all the frames, which has three categories: checkerboard, car, and people sequences. Since checkerboard sequences do not correspond to natural scenes. We just use one sequence (1R2TCR) to show the effectiveness of our method in dealing with cameras rotation. The Berkeley dataset is derived from the Hopkins dataset, which consists of 26 moving camera videos: car, people, and Marple sequences. This dataset has full pixel-level annotations on multiple objects for a few frames sampled throughout the video. Since Marple sequences mainly contain static scenes or the objects are static, it is little challenging for our method. Thus, they are not involved in the experiments. Complex Background and SegTrack datasets contain extremely challenging scenes, where the background motion is much more complex than

Fig. 3 Scenario 1: cars2 sequence. (a) The sequence from frame 1 to frame 20. (b) An optical flow from two adjacent frames. (c) Potential E . (d) Potential $\tilde{W}$. (e) The Incoherence Map (IM). (f) Progressive Quad-Tree partition on IM. (g) The approximated inlier potential using low order polynomial. (h) The estimated inlier flow field. (i) The potential of outliers. (j) The recovered motion flow of moving objects. (k) The segmentation result.

Fig. 4 Scenario 2: checkerboard sequence. Please refer to Fig. 3 for the description of subfigures.

other datasets. They also provide full pixel-level annotations on multiple objects at each frame within each video. These two datasets are employed to highlight the advantage of our method. FBDynScn video data contain much dynamic textures which other datasets do not have. We involve this dataset to illustrate the robustness of the proposed method on dynamic texture scenes.

We first illustrate the performance of the proposed method on three sequences that consist of camera translation, rotation and zoom in/out, respectively. Then we compare our approach with methods using optical flow [1], [2], [4], [5], [6], [7] and trajectories clustering [7], [11], [24], [26] on all four datasets. Optical flow is calculated using Brox's method [27] and optimized by [28].

5.1 Performance on Three Typical Sequences

We demonstrate the performance of our method on three typical Scenarios: *cars2*, *1R2TCR*, and *drive* that involve varied camera motions. To clarify the motions involved in the video, we add a prefix before the video name. Specifically speaking, each moving object in the scene is indexed by a number (e.g. 1, 2, ...), the camera is denoted by 'C', each motion type of objects and camera is indicated by a letter: 'R' for rotation, 'T' for translation, and 'P' for radial motion. For example, 1R2TCRT means that the

first moving object rotates, the second translates, and the camera motion consists of both rotation and translation.

[1] 1T2T3TCT-Cars2 Sequence: This sequence is from the Berkeley dataset, please see Fig. 3. Three cars are translating in the scene: one is near the center of the view, another is on the left side near to the image boundary, and the third one is on the right side partially occluded by the white car. The camera is translating. (a) The frame 1 to frame 20 from the video. (b) One optical flow. We can see that only the motion flow of the left car is very obvious, whereas the others are drowned out by the inlier. Two potentials (c) and (d) show the curl and divergence components after applying amended HHD on optical flow. (e) The Incoherence Map depicts all three moving cars clearly. (f) The progressive Quad-Tree labels outliers. (g) The approximated inlier potential using low order polynomial. (h) The estimated inlier flow field. It is quite coherent. The motion flows of moving objects are recovered accurately in (i) and (j). This example demonstrates a very common 3D motion segmentation scenario.

[2] 1R2TCR-Checkerboard Sequence: This data is from the Hopkins dataset, please see Fig. 4. This scene involves three motions: a rotation view (camera motion), in which a basket is rotating and a box is translating from left to right. We use this sequence to demonstrate the performance of our method in dealing with camera rotation. (a) Checkerboards are placed in the scene

Fig. 5 Scenario 3: Drive sequence. Please refer to Fig. 3 for the description of subfigures.

Table 1 F-measures of six other methods using optical flow and ours.

Sequences	GME-SEG	LS	GD	Filter	RANSAC	FOF	FOR+color	Our
Cars1	78.33	86.18	18.01	82.87	60.42	47.81	50.84	76.38
Cars2	55.90	65.97	15.70	78.28	34.21	46.37	56.60	83.23
Cars3	65.21	79.43	22.29	74.56	35.80	67.18	73.57	87.47
Cars4	45.69	49.78	22.96	77.22	22.81	38.51	47.96	84.52
Cars5	54.67	61.93	33.08	81.17	25.24	64.85	70.94	84.92
Cars6	33.01	51.23	28.77	57.53	13.40	78.09	84.34	85.81
Cars7	37.89	36.36	36.92	60.47	13.79	37.63	42.92	86.10
Cars8	62.20	81.24	8.57	78.44	37.02	87.13	87.61	90.78
Cars9	72.99	80.99	17.97	68.19	54.69	68.99	66.38	77.52
Cars10	60.01	66.04	14.34	90.95	81.78	53.98	50.84	54.93
People1	34.11	38.32	40.76	71.84	12.06	56.76	69.53	80.14
People2	78.16	84.45	69.30	81.70	37.53	85.35	88.40	89.91
drive	8.14	6.30	41.18	32.40	5.76	30.13	61.80	83.03
forest	15.42	11.01	15.41	19.87	10.67	19.48	31.44	35.81
parking	33.20	21.57	43.84	62.29	17.47	43.47	73.19	83.57
store	14.39	10.94	32.10	29.32	9.68	28.46	70.74	80.10
traffic	14.77	15.80	34.55	15.04	15.49	66.08	71.24	71.77
birdfall2	9.39	3.84	0.99	64.00	3.25	68.68	75.69	76.23
girl	22.51	20.26	15.36	18.21	12.33	75.73	81.95	78.06
parachute	23.01	18.97	12.88	16.30	44.03	51.49	54.36	86.72
cheetah	21.33	14.93	43.59	12.05	9.85	12.68	22.31	55.77
penguin	10.34	18.84	15.34	5.53	18.66	14.74	20.71	21.71
monkeydog	22.29	20.74	16.46	18.93	12.31	10.79	18.62	45.45

to obtain strong features (corners and edges) for the feature based methods to segment motions. Our method, in contrast, does not rely on these features. From the optical flow (b), it is impossible to figure out what exact motions are involved in the scene. (e) The IM, however, has showed the outliers, and the progressive Quad-Tree labels them well in (f). From the approximated inlier potential (g) and its motion flow field (h), we can clearly find the inlier is rotation. This instance demonstrates the high performance in dealing with camera rotation.

[3] **1TCP-Drive sequence:** This sequence is from the Complex Background dataset, please see Fig. 5. We use this sequence to demonstrate the performance of our method in dealing with camera zoom in/out. It involves two motions: a camera is moving towards resulting a radial flow field. A car is turning to left at the corner generating a translation. The curl-free potential (c) depicts the most inlier. Finally, the radial flow field is approximated in the inlier potential (g) and estimated in (h) clearly. The IM (e) depicts, and the progressive Quad-Tree (f) locates outlier successfully. The estimated inlier (h) and the segmentation result demonstrate the effectiveness of our method in dealing with camera zoom in/out.

5.2 Comparison with State-of-the-arts

We compared the proposed method with the well-known and latest methods in two categories.

- **Using optical flow.** We select six recent developed methods which are: joint inlier estimation and segmentation (GME-SEG) [2], iterative estimation using least-square (LS) [6], and gradient descent (GD) [5], outlier rejection filter (Filter) [1], RANSAC [7], and the latest developed FOF [4] that presents two versions: (1) FOF, which uses optical flow information only, and (2) FOF+color+prior, which combines optical flow, color appearance and a prior model to improve the accuracy. The first five methods are implemented using the source code provided by [2]. The performances of FOF and FOF+color+prior is presented from [4] directly.
- **Trajectories clustering.** Three well-known methods are selected: GPCA with spectral clustering [11], Local Subspace Affinity (LSA) [26] and RANSAC [7]. The source codes of these methods are provided in Ref. [24].

We employ the *F-measure* to evaluate the performance of each algorithm. *F-measure* [29] is a more unbiased evaluation for binary classification problem because it considers both the precision P_r and the recall R_c of the test to compute the score as:

Fig. 6 Segmentation results of six other methods using optical flow and ours on challenging scenes: (a) input videos, from top to bottom: cars2, people2, forest, store, parachute, traffic, (b) segmentation by GME-SEG [2], (c) LS [6], (d) GD [5], (e) Filter [1], (f) RANSAC [7], (g) FOF [4], (h) FOF+color+prior [4], (i) our result, (j) ground truth.

Table 2 F-measures of three trajectories clustering methods and ours.

Sequences	GPCA	RANSAC	LSA	Our
1R2TCR	84.93	92.64	97.05	97.90
Cars1	100	100	98.09	99.35
Cars2	84.89	92.91	92.19	96.55
Cars3	93.27	94.55	52.15	96.30
Cars4	100	100	98.85	100
Cars5	45.88	88.89	63.71	98.16
Cars6	94.12	94.12	92.31	96.00
Cars7	83.33	83.33	78.00	83.33
Cars8	99.39	96.89	98.78	98.77
Cars9	61.18	59.57	49.02	72.73
Cars10	52.83	53.46	52.49	90.63
People1	91.53	91.53	91.53	96.43
People2	100	84.75	97.96	100

$$F = \frac{2 \times R_c \times P_r}{R_c + P_r}.$$

The numerical comparison results are shown in **Table 1** and **Table 2**, respectively. Table 1 reports the F-measure of ours and other six methods using optical flow on three benchmark datasets: Berkeley, Complex Background, and SegTrack. Table 2 reports the F-measure of ours and those three trajectories clustering methods on Hopkins dataset. We must clarify that the calculations of F-measure in two categories are bit different. The methods using optical flow perform segmentation on each pixel, thus, F-measure is computed using all pixels on the field. Nevertheless, trajectories clustering methods track key feature points to obtain trajectories, then, only feature points are considered in the F-measure. That is the reason that the proposed method has higher scores in Table 2 than those in Table 1 on the same video sequence.

5.2.1 Comparison with Methods Using Optical Flow

Table 1 shows that our method outperforms others and the lat-

est FOF and FOF+color+prior on most videos: it raises the F-measure by 10%–30% on *Cars 2, 3, 4, 7*, and *People 1* sequence in the Berkeley dataset; around 10% on the *drive*, *parking*, and *store* sequences in the Complex Background dataset; and more than 20% on the *parachute* and *monkeydog* sequences in the SegTrack dataset. This result is quite appealing even on videos containing extremely challenging scenes, such as the ones with occlusions, complex backgrounds, and noises. The visual results are shown in **Fig. 6** as well, where the last column is the ground truth. In most cases, our segmentation agrees with the ground truth more than the existing methods.

Table 1 also shows that our method performs a bit worse on the *cars 1, 9, 10*, and *girl* sequences. The reasons are that the *cars 1, 9* and *10* sequences contain some weak or analogous moving objects, whose motions are inconspicuous in the camera motion field. For example, a big truck is in cars 10 sequence, which appears almost static on many frames. When these motions are mixed with the inlier, the entire motion field is rather coherent. They, therefore, cannot rest on the IM, which makes our method fail to detect outliers. Moreover, we have to point out that the accuracy of the optical flow affects the performance of our method as well. The *girl* sequence shows such an example. It captures a fast running girl in the sports yard. Some frames are blurred terribly, and have severe noisy optical flows. In this case, only optical flow is not sufficient. That's why FOF+color+prior utilizes additional information (color appearance) and prior models to improve the performance. In addition, all methods appear less accurate on the three sequences (*cheetah*, *penguin*, *monkeydog*) in the SegTrack dataset, because they have multiple moving objects, but the ground truth intended for tracking one primary object as the foreground, causing all methods appear less accurate.

Table 3 F-measure of FBM and ours on FBDynScn dataset.

Sequences	FBM	Our
Sailing2	49.45	75.75
Boat1Person1	68.49	68.95
Fountain2Person2	56.61	76.03
Person2Tree1	37.28	54.59
Boat2	74.09	57.17

5.2.2 Comparison with Trajectories Clustering Methods

These methods analyze trajectories using multiple frames, and some need special initialization at the first frame. Theoretically speaking, they should not be directly comparable to inlier and outliers accuracy measure. Our method is based on optical flow, and a frame-to-frame method requiring neither initialization nor prior knowledge. To demonstrate the performance with a persuasive result, we compare our method with three trajectories clustering methods [7], [11], [24].

Table 2 shows that our method performs better than or equal to these methods on most sequences (*IR2TCR*, *Cars* 2, 3, 4, 5, 6, 7, 9, and *People* 1). On sequences (*Cars* 1, 8), they slightly outperform ours. That's because a special initialization procedure is required to segment the objects in the first frame. Our method, however, has no such assumption. For example, the optical flows of first a few frames in *Cars* 1 video are pretty coherent, our method fails to detect the object. As mentioned in the related works, we argue that trajectories clustering methods suffer from several limitations: (1) they rely much on the existence of strong key features in the scene and depend on a robust feature extraction, tracking, and classification algorithms; (2) they are very sensitive to the incomplete trajectories and tracking errors caused by occlusion problems and noises; (3) The segmentation of sparse features is difficult to reveal the entire motion region in details. In contrast, our method has no above limitation, and is more robust to cope with complex scenes with various camera motions, even when occlusions and noises present in the scene. In addition, our method preserves the shape of moving objects well and recovers their true motions, which are much valuable in vision based applications.

One may also consider the joint foreground-background mixture model (FBM) [20] as a kind of trajectories clustering method. Because it divides video into spatiotemporal overlapping patches, but encodes the appearance of each patch at time t by location-specific dynamic texture (DT) components and global DT components. Therefore, it is able to well estimate the background with DT, and segment motions. We also test our method on the FBDynScn dataset [20] which contains the dynamic texture scenes, and compare it with FBM. Actually, if the DT is weak in the video, our method treats it as inlier and produce a reasonable segmentation result. Table 3 and Fig. 7 show that our method outperforms FBM on *Sailing2*, *Fountain2Person2*, *Person2Tree1* sequences, even these videos contain a lot of dynamic texture (e.g., spoondrift, fountain, waving leaves), and raise the F-measure 17%~25% up. However, our method performs similarly as FBM on *Boat1Person1*, and worse on *Boat2*. The reason is that the boats in these two videos move extremely slow, which is almost coherent with the inlier in optical flow. Thus, the boat motion does not appear in the IM. In this case, the appearance feature in

Fig. 7 Segmentation results on the FBDynScn dataset and comparison with the state-of-the-art FBM: (a) input sequences, from top to bottom: Sailing2, Boat1Person1, Fountain2Person2, Person2Tree1, Boat2, (b) ground truth, (c) FBM [20], (d) our result.

Fig. 8 Three failure cases: (a) The first one; (b) the second one; (c) the third one.

FBM is a plus in segmentation task.

5.2.3 The Failure Cases

Although, the proposed method achieves inspiring performance, extensive experiences shows it may fail in the following three cases:

- (1) Motions of moving objects are very weak comparing with the camera motion, see Fig. 8(a). In this case, the outliers are more likely to be decomposed by HHD because they are pretty coherent with the inlier, and cannot appear in IM. Such as the *cars* 1, 9 and 10 sequences, *forest*, *Boat1Person1*, and *Boat2* contain some object motions which are very small.
- (2) A few isolated static objects stand alone in a texture-free background while camera is moving, see Fig. 8(b). In this case, our method may mistake these isolated static objects as moving objects.
- (3) The outlier is greater than inlier. Figure 8(c) shows a scenario where the bus occupies the most view. In this case, HHD fails to decompose the inlier because of the global minimization.

6. Conclusions

We have presented a general framework for 3D motion segmentation on a wide bank of moving camera videos. This framework solved two problems: 1) the amended HHD can handle the coherent inlier of all three primary motion flows (translation, ro-

tation, and radial flow), 2) the proposed Incoherence Map and progressive Quad-Tree precisely label the outliers, motion discontinuities and noise. The afterward inlier estimation is achieved by approximating low order polynomials using the rest samples on two potentials in HHD. This compensates the depth discontinuity in the 3D motion. We have evaluated our approach on four benchmark datasets. Extensive experiments showed performance comparable to state-of-the-art methods. In the future work, we will show that more coherent information (e.g., colors, the direction only) might further help our method for segmentation.

Acknowledgments We are very thankful to the review committee for their contributions in correcting the errors in the writing. This work is supported by: Japan Society for the Promotion of Science, Grants-in-Aid for Scientific Research C (No. 15K00236).

References

- [1] Chen, Y.M. and Bajic, I.V.: Motion vector outlier rejection cascade for global motion estimation, *IEEE Signal Processing Letters*, Vol.17, No.2, pp.197–200 (2010).
- [2] Chen, Y.M. and Bajic, I.V.: A joint approach to global motion estimation and motion segmentation from a coarsely sampled motion vector field, *IEEE Trans. Circuits System and Video Technology*, Vol.21, No.9, pp.1316–1328 (2011).
- [3] Qian, C. and Bajic, I.V.: Global motion estimation under translation-zoom ambiguity, *Proc. IEEE PacRim*, pp.46–51 (2013).
- [4] Narayana, M., Hanson, A. and Learned-Miller, E.: Coherent motion segmentation in moving camera videos using optical flow orientations, *ICCV* (2013).
- [5] Su, Y., Sun, M.-T. and Hsu, V.: Global motion estimation from coarsely sampled motion vector field and the applications, *IEEE Transactions on Circuits System and Video Technology*, Vol.15, No.2, pp.232–242 (2005).
- [6] Smolic, A., Hoeyneck, M. and Ohm, J.-R.: Low-complexity global motion estimation from P-frame motion vectors for MPEG-7 application, *ICIP*, pp.271–274 (2000).
- [7] Fischler, M. and Bolles, R.: RANSAC random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography, *Comm. ACM*, Vol.26, pp.381–395 (1981).
- [8] Zografos, V. and Nordberg, K.: Fast and accurate motion segmentation using Linear Combination of Views, *BMVC* (2011).
- [9] Aldroubi, A.: A review of subspace segmentation: Problem, nonlinear approximations, and applications to motion segmentation, *ISRN Signal Process*, pp.1–13 (2013).
- [10] Ichimura, N.: Motion segmentation based on factorization method and discriminant criterion, *ICCV*, pp.600–605 (1999).
- [11] Vidal, R., Ma, Y. and Soatto, S.: Two-view multibody structure from motion, *International Journal of Computer Vision (IJCV)*, Vol.68, No.1, pp.7–25 (2006).
- [12] Sugaya, Y. and Kanatani, K.: Geometric structure of degeneracy for multi-body motion segmentation, *Workshop on Statistical Methods in Video Processing*, pp.1–2 (2004).
- [13] Costeira, J. and Kanade, T.: A multibody factorization method for independently moving objects, *International Journal of Computer Vision (IJCV)*, Vol.29, No.3, pp.159–179 (1998).
- [14] Kanatani, K. and Matsunaga, C.: Estimating the number of independent motions for multibody motion segmentation, *ECCV*, pp.25–31 (2002).
- [15] Brox, T. and Malik, J.: Object segmentation by long term analysis of point trajectories, *ECCV* (2010).
- [16] Ochs, P., Malik, J. and Brox, T.: Segmentation of moving objects by long term video analysis, *IEEE Trans. Pattern Analysis and Machine Intelligence (TPAMI)* (2014).
- [17] Elqursh, A. and Elgammal, A.: Online Moving Camera Background Subtraction, *ECCV*, pp.228–241 (2012).
- [18] Kwak, S., Lim, T., Nam, W., Han, B. and Han, J.H.: Generalized background subtraction based on hybrid inference by belief propagation and Bayesian filtering, *ICCV*, pp.2174–2181 (2011).
- [19] Shi, F., Zhou, Z., Xiao, J. and WeiWu: Robust Trajectory Clustering for Motion Segmentation, *ICCV* (2013).
- [20] Adeel, M., Zhang, W. and Chan, A.B.: Joint Motion Segmentation and Background Estimation in Dynamic Scenes, *CVPR*, pp.368–375 (2014).
- [21] Irani, M., Rousso, B. and Peleg, S.: Recovery of ego-motion using image stabilization, *CVPR* (1994).
- [22] Helmholtz, H.: On Integrals of the Hydrodynamical Equations, which Express Vortex-Motion, *Philosophical Magazine and J. Science*, Vol.33, No.226, pp.485–512 (1867).
- [23] Bhatia, H., Norgard, G., Pascucci, V. and Bremer, P.-T.: The Helmholtz-Hodge Decomposition — A Survey, *IEEE Trans. Visualization and Computer Graphics (TVCG)*, Vol.19, No.8, pp.1386–1404 (2013).
- [24] Tron, R. and Vidal, R.: A benchmark for the comparison of 3D motion segmentation algorithms, *CVPR* (2007).
- [25] Tsai, D., Flagg, M. and M. Rehg, J.: Motion Coherent Tracking with Multi-label MRF optimization, *BMVC* (2010).
- [26] Yan, J. and Pollefeys, M.: A general framework for motion segmentation: Independent, articulated, rigid, non-rigid, degenerate and nondegenerate, *ECCV*, pp.94–106 (2006).
- [27] Brox, T., Bruhn, A., Papenberger, N. and Weickert, J.: High accuracy optical flow estimation based on a theory for warping, *ECCV*, pp.25–36 (2004).
- [28] Liang, X., McOwan, P. and Johnston, A.: A Biologically Inspired Framework for Spatial and Spectral Velocity Estimations, *Journal of the Optical Society of America A*, Vol.28, No.4, pp.713–723 (2011).
- [29] Dembczynski, K., Jachnik, A., Kotlowski, W., Waegeman, W. and Hullermeier, E.: Optimizing the F-measure in multi-label classification: Plug-in rule approach versus structured loss minimization, *International Conference on Machine Learning (ICML)*, pp.1130–1138 (2013).

Xuefeng Liang received his M.S. degree in computer science from Xidian University, China, in 2003, Ph.D. degree in information science from Japan Advanced Institute of Science and Technology, Japan, in 2006. Afterwards, he held JST research fellowship in Ubiquitous Functions Research Group, ISRI, National Institute of Advanced Industrial Science and Technology in Japan. From 2008 to 2010, he worked as research assistant in both University College of London and Queen Mary University of London. He is currently an associate professor in Kyoto University. He is on the editorial board of two international journals, and has served as TAP and TCP members for several international conferences (i.e., ACPR 2015). His research interests include visual perception, computer vision, algorithms and biometrics.

Cuicui Zhang received her B.Eng. degree in Software Engineering and M.Sc. degree in Computer Science from Shandong University, China, in 2008 and 2011, respectively. She got her Ph.D. degree in information science from Kyoto University, Japan, in 2015. Currently, she is with Tianjin University, China. Her research interests include Face Recognition, Human Emotion Perception, Motion Analysis and Computer Graphics.

Takashi Matsuyama received his B. Eng., M. Eng., and D. Eng. degrees in electrical engineering from Kyoto University, Japan, in 1974, 1976, and 1980, respectively. He is currently a professor in the Department of Intelligence Science and Technology, Graduate School of Informatics, Kyoto University. He served

as the director of the Academic Center for Computing and Media Studies (2002–2006), the director general of the Institute for Information Management and Communications (2005–2010), and a vice president of Kyoto University (2008–2010). He has been studying cooperative distribute sensing-control-reasoning systems over 30 years. Their application fields include knowledge-based image understanding, visual surveillance, 3D video, human-computer interaction, and smart energy management. He wrote more than 100 journal papers and more than 20 books including three research monographs: *A Structural Analysis of Complex Aerial Photographs*, PLENUM, 1980, *SIGMA: A Knowledge-Based Aerial Image Understanding System*, PLENUM, 1990, and *3D Video and its Applications*, Springer, 2012. He organized and served as the leaders of 45-years national research projects and won more than 20 best paper awards from Japanese and international academic societies including the Marr Prize at ICCV'95 and the commendation for Science and Technology by the Minister of Education, Culture, Sports, Science and Technology in 2009. He is on the editorial board of the *Pattern Recognition Journal*. He was awarded Fellowships from the International Association for Pattern Recognition, the Information Processing Society of Japan, and the Institute of Electronics, Information, and Communication Engineers Japan.

(Communicated by *Ian Reid* and *Hideo Saito*)