This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2017.2763602, IEEE Transactions on Circuits and Systems for Video Technology

Sparse-to-Dense Depth Estimation in Videos via High-Dimensional Tensor Voting

Botao Wang, Junni Zou, *Member, IEEE,* Yong Li, Kuanyu Ju, Hongkai Xiong, *Senior Member, IEEE,* Yuan F. Zheng, *Fellow, IEEE*

***Abstract*—Due to the popularity of three-dimensional videos, 2D-to-3D video conversion has become a hot research topic for the past few years. The most critical issue in 3D video synthesis is the estimation of depth maps for the video frames. Numerous efforts have been devoted in fully automatic and semi-automatic depth estimation approaches, although the discontinuity of depth field and the ambiguity of motion boundary are still the main challenges in depth estimation. This paper proposes a semi-automatic structure-aware sparse-to-dense depth estimation method, which leverages the tensor voting at two different levels to propagate depth across frames. In the first level, a 4D tensor voting is performed to remove outliers caused by inaccurate motion estimation. Noticing that the 4D tensors of correctly matched points should lie on the smooth layer in the manifold, we utilize the variety saliency defined by the eigen-system of the tensor for outlier removal. In the second level, a high-dimensional tensor voting algorithm, incorporating spatial location, motion and color into the tensor representation, is devised to propagate the depth from the sparse points to the entire image domain. By projecting the input feature into the tangent space, the relation between the location, motion, color and the depth can be established by voting process. Extensive experiments on public dataset validate the effectiveness of the proposed method in comparison with state-of-the-art depth estimation approaches.**

***Index Terms*—Depth estimation, tensor voting, motion estimation, bilateral filtering.**

I. Introduction

Three dimensional (3D) video can provide an enhanced visual experience with depth perception beyond conventional 2D contents. With the growth of 3D display devices, the increasing demand for 3D contents has aroused a significant challenge to the 3D industry. A promising way is to produce new 3D videos from massive existing monocular 2D videos. A typical 2D-to-3D conversion process consists of two steps: depth estimation and depth-based rendering. Depth estimation is a critical issue because without accurate depth, synthesized stereo views cannot be well generated by depth-based rendering (e.g. DIBR [1, 2]).

Existing 2D-to-3D techniques can be divided into two categories: fully automatic methods (e.g., [3]) and semi-automatic methods (e.g., [4, 5]), depending on whether user interactions are involved in depth estimation. Fully-automatic methods are limited to some restricted scenarios, thus do not work well for arbitrary scenes. In contrast, semi-automatic methods can balance 3D content quality with production cost, which makes them more effective and flexible. Aiming at desirable 3D quality, semi-automatic methods require skilled operators to assign depth to the key frames in 2D videos. Later, the depth information can be propagated automatically from the key frames to non-key frames over the entire video sequence. Depth propagation is a major part of depth estimation, thereby playing a key role in the semi-automatic group. It should be noted that the proposed method belongs to this category.

Bilateral filtering has been adopted by many approaches [4, 6] to propagate depth across frames. The main drawback of bilateral filtering is that it is sensitive to occlusion and inaccurate motion estimation. In comparison, the proposed method utilizes high-dimensional tensor voting for depth estimation, which is robust against occlusion by propagating depth along the structure of the same object. To estimate motion, many approaches [7, 8] apply bi-directional matching of optical flow. However, they often suffer from over-smooth and motion discontinuity. On the contrary, the proposed method does not estimate motion densely, but only for a sparse set of feature points, whose motions are easy to determine, and use 4D tensor voting to remove unreliable matches.

The contribution of this paper is two-fold. First, a robust sparse depth map estimation method is proposed, which is illustrated in the red block in Fig. 1. The depth values of a sparse set of interest points can be estimated in three steps, namely, *interest point extraction*, *motion estimation* and *outlier removal*. Specifically, the first step extracts a set of interest points from the image, which can be accurately matched across frames, by corner detection and uniform sampling to ensure the distinctiveness and the spatial coverage. Next, the motion of the interest points between two frames will be estimated by optical flow tracking, so that the depth can be propagated in accordance with the motion. Finally, since the motion estimation is not always reliable, an outlier removal procedure using 4D tensor voting is designed to eliminate the mismatched interest points. By examining the variety saliency of the 4D tensors, the proposed algorithm is more effective in outlier detection compared with conventional bi-directional motion validation.

Second, based on the depth of the sparse set of interest points, a depth interpolation algorithm is contrived, which is illustrated in the green block in Fig. 1. High-dimensional tensor voting [9, 10] is leveraged to reliably propagate the depth of the interest points to the entire image. To be concrete, the proposed algorithm encodes each point in the depth field with the spatial coordinates, motion, color and depth for reliable depth inference. To extract local structures from the manifold that the points lies on, the tangent space spanned by the

B. Wang, Y. Li, K. Ju and H. Xiong are with the Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai, 200240, China. Email: {botaowang, marsleely, xionghongkai}@sjtu.edu.cn.

J. Zou is with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240. Email: zoujn@cs.sjtu.edu.cn

Y. Zheng is with the Department of Electrical and Computer Engineering, The Ohio State University, Columbus, OH 43210. Email: zheng@ece.osu.edu.

This work was supported in part by NSFC under Grant 61425011, Grant 61529101, Grant 61720106001, Grant 61622112, Grant 61472234, and in part by the Program of Shanghai Academic Research Leader under Grant 17XD1401900.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2017.2763602, IEEE Transactions on Circuits and Systems for Video Technology

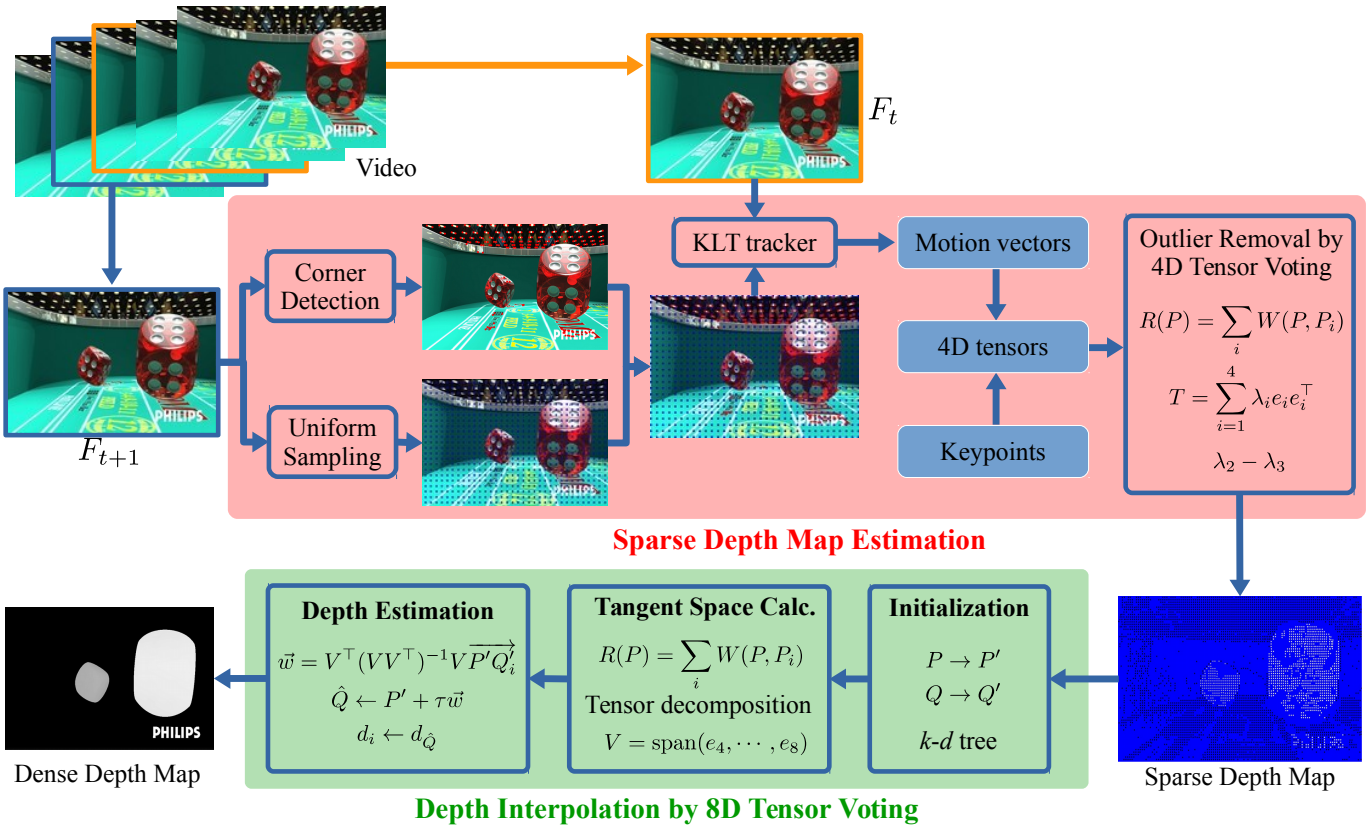


Fig. 1. The overall framework of the proposed method.

tangent vectors of the free parameters is utilized to establish the relation between the visual and motion input to the depth. Comprehensive experiments on public dataset validate the effectiveness of the proposed method in comparison with state-of-the-art methods in depth estimation for videos.

The rest of this paper is organized as follows: Section II reviews related literature; Section III describes the proposed sparse-to-dense depth estimation approach with high-dimensional tensor voting. Section IV evaluates the proposed method through extensive experiments; Finally, Section V concludes the paper.

II. Related Work

Existing 2D-to-3D techniques can be divided into two categories: automatic and semi-automatic methods. The former can estimate the depth of a video sequence automatically without the human intervention. Depth is estimated from monocular depth cue, such as shape-from-shading [11] and depth-from-defocus [12]–[14]. However, these methods can only extract 3D information from constrained scenes. Multi-view stereo methods attempt to recover depth by estimating 3D information from images of the same scene captured in different time, e.g. structure-from-motion [15]–[17]. Recently, learning-based methods have been exploited to generate depth map of a monocular image. Konrad et al. [18] presented two learning-based 2D-to-3D conversion schemes. The first utilizes low-level video attributes to estimate depth value of a pixel by learning a local point transformation, while the second estimates the global depth map of a query image from a repository of image-depth pairs using nearest-neighbor search. Wang and Jung [7] proposed example-based video stereolization with foreground segmentation and depth propagation. Although fully automatic approaches minimize human involvement, their performance and use scenarios are very limited. In comparison, the proposed method is capable of producing high-quality depth maps with the guidance of cheap sparse key-frames in a semi-automatic way.

Semi-automatic methods are more successful in producing high stereo quality under the guidance of human intervention. Guttmann et al. [5] developed a dense depth fusion method based on user scribbles. A classifier is trained to predict the disparity based on the user scribbles in some frames, and the disparity of entire shot can be recovered by an optimization process, which is constrained by the original scribbles and the high confidence predictions. 3D tensor voting is utilized in [19] for multiview stereo reconstruction to enforce photoconsistency, visibility and geometric consistency. A semi-automatic 2D-to-3D conversion system in [6] contains two main steps: depth assignment in key frames and dense depth propagation from key frames to non-key frames. Users draw some simple strokes to assign depth values in key frames, and the unmarked areas are assigned using graph-cut optimization. The depth propagation for non-key frames is based on shifted bilateral filtering. However, the critical limitation of methods based on user strokes is that they only work for simple videos, because users can not possibly label every object in

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2017.2763602, IEEE Transactions on Circuits and Systems for Video Technology

the frame by hand. On the contrary, the proposed method can generate accurate depth maps for complicated videos. To improve the efficiency of depth propagation, Li et al. [8] propagated depth for non-key frames via bi-directional motion estimation, where bi-directional motion vectors are estimated to determine the depth propagation strategy. [20] extended the depth propagation algorithm, and proposed a depth refinement process based on the natural scene statistics (NSS) model.

In [4], depth is attained by bilateral filtering and refined through a block-based motion compensation from previous frames. In 2011, the bilateral filtering based method was extended in [6], where the depth map is propagated by shifted bilateral filtering with motion information. Bilateral filtering based approaches are sensitive to occlusions and inaccurate motion estimation, while the proposed method is robust against occlusion and motion via high-dimensional tensor voting. Li et al. [8] propagated depth for non-key frames via bi-directional motion estimation, where bi-directional motion vectors are estimated to determine the depth propagation strategy. In [7], motion vectors are estimated by the Horn-Schunk optical flow estimation. To alleviate error propagation, post-filtering is performed before estimating depth to the next frame. It should be noted that these methods suffer from over-smooth and motion discontinuity, but the proposed method yields accurate motion for a sparse set of keypoints.

III. Sparse-to-Dense Depth Estimation via High-Dimensional Tensor Voting

A. Framework Overview

The overall architecture of the proposed method is illustrated in Fig. 1. The video sequence is denoted by $\{F_t\}_{t=1}^{K}$, where K is the number of frames and F_t is the t-th frame. The depth map of the first frame is given and denoted by D_1, and the depth maps of the remaining frames, i.e., $\{D_t\}_{t=2}^{K}$, will be estimated by propagating the depth along the time line from the first frame. In other words, the depth map D_{t+1} will be estimated based on F_t, F_{t+1} and D_t. In specific, the proposed method propagates the depth from the current frame to the next frame in two steps: *depth map initialization by sparse motion estimation* and *depth interpolation by high-dimensional tensor voting*.

In sparse motion estimation, a set of interest points are extracted from F_t, and the corresponding points in F_{t+1} will be attained by visual tracking. To rule out the false matches from tracking, an outlier removal procedure is performed to retrieve high-confident points for sparse depth estimation and depth propagation. It is worth mentioning that, on the contrary to conventional approaches that use forward-backward motion verification to remove false matches, the proposed method investigates 4D tensor voting based on the location and the motion of the interest points for outlier removal, which is more effective in capturing the geometric structure of the depth map. Hence, the depth of the sparse set of keypoints in D_{t+1} can be obtained by propagating the depth from D_t along the motion vectors.

In depth interpolation, the entire depth map of frame F_{t+1} will be estimated with high-dimensional tensor voting on the basis of the depth of the initial points. To obtain a reliable representation, each initial point is encoded by its location, motion, color and depth. Then, each point propagates its information to its neighbors via tensor voting, and the tangent space spanned by the free parameters of the tensors will be derived for depth inference. Hence, the depth of a point can be interpolated by iteratively moving towards the desirable direction based on the tangent space.

As follows, we will revisit the basic concepts about tensor voting in Section III-B, and then describe the technical details on the depth map initialization by sparse motion estimation in Section III-C and depth interpolation by high-dimensional tensor voting in Section III-E.

B. Preliminaries in Tensor Voting

In this section, we briefly review the necessary notations and formulations in tensor voting. More technical details can be found in [21], and a comprehensive survey is available in [10]. Two special cases of tensors are: *stick tensor* and *ball tensor*. The stick tensor has only one non-zero eigenvalue and represents perfect certainty for a hyperplane normal to the eigenvector that corresponds to the non-zero eigenvalue. On the other hand, all eigenvalues of a ball tensor are identical and non-zero, which represents perfect uncertainty in orientation, or, just the presence of an unoriented point.

After the data points are represented with tensors, they can refine the information they carry based on their neighbors through a voting process. The vote that the voter casts to the receiver has the orientation the receiver would have, if both the voter and receiver belong to the same structure. The magnitude of the vote is proportional to the confidence that the voter and receiver belong to the same structure.

C. Depth Map Initialization by Sparse Motion Estimation

To obtain an accurate depth map, the depth of a sparse set of reliable and discontinuity-preserving interest points is estimated in the first step, which will be propagated to other points in the second step.

The interest point are sampled from F_{t+1} in two ways: non-uniform sampling from corners, and uniform sampling from regular grids. In concrete, to obtain an accurate matching of the interest points in F_t and F_{t+1}, the initial points are supposed to be distinct, so that they can be extracted repetitively and matched unambiguously. Hence, we extract Shi-Tomasi corners [22] from the image, which have been proven to be effective in various computer vision tasks, e.g., visual tracking, image retrieval and object detection. On the other hand, since the distribution of corner points are highly imbalanced, we uniformly sample the interest points in the regular grids at a stride of 4 pixels over the entire image. Hence, the initial points are capable of capturing the main structures of the image. In addition, due the the fact that points in textureless region may result in large error in visual tracking, we remove the points of small eigenvalue of the structure tensor. Eventually, the initial points extracted from F_{t+1}, consisting of points from the corners and from the textureless region, can be denoted by $\mathcal{P}_{t+1} = \{p_i^{t+1}\}_{i=1}^{M}$,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2017.2763602, IEEE Transactions on Circuits and Systems for Video Technology

TABLE I
NOTATIONS

K	Number of frames	$F_t, t = 1, \cdots, K$	The t-th frame
D_t	depth map of frame t	$p = (x, y)$	Coordinates of interest point
$\mathcal{P} = \{p_i\}_{i=1}^{M}$	Set of interest points	M	Number of interest points
$u^{t+1} = (x^t - x^{t+1})$	Horizontal motion vector	$v^{t+1} = (y^t - y^{t+1})$	Vertical motion vector
T	N-dimensional tensor	$e_i, i = 1, \cdots, N$	Eigenvectors
$\lambda_i, i = 1, \cdots, N$	Eigenvalues	$\mathcal{N}(P)$	Set of neighbors of P
$R(P)$	Votes received by P	$W(P, P_i)$	Vote P_i casts to P
(r, g, b, d)	RGB intensities and depth	V	Tangent space
$\vec{w}$	Projected direction		

where $\mathcal{P}_{t+1}$ is the point set in F_{t+1}, M is the number of initial points, and $p_i^{t+1} = (x_i^{t+1}, y_i^{t+1})$ is the coordinates of the i-th points.

To estimate the displacement of the initial points, KLT tracker [22] is leveraged to find the corresponding points of $\mathcal{P}_{t+1}$ in F_t, which is denoted by $\mathcal{P}_t = \{p_i^t\}_{i=1}^{M}$, where p_i^t is the location of p_i^{t+1} in F_t. Consequently, the motion vectors of the initial points are

$$u_i^{t+1} = x_i^t - x_i^{t+1}, \quad v_i^{t+1} = y_i^t - y_i^{t+1}, \quad i = 1, \cdots, M, \tag{1}$$

where u_i^{t+1} and v_i^{t+1} are the horizontal and vertical displacements of p_i^{t+1} with respect to p_i^t.

The examples of the initial points extracted from the test sequences are demonstrated in Fig. 2, where the blue points are uniformly sampled from regular grids over the image, and the red points are extracted from corners.

D. Outlier Removal by 4D Tensor Voting

To eliminate the wrong matches obtained from visual tracking, a 4D tensor voting process is performed. The use of a voting process for feature inference from sparse and noisy data was introduced by Guy and Medioni [23] and then formalized into a unified tensor framework. This methodology is non-iterative and robust to considerable amounts of outlier noise. The only free parameter is the scale of analysis, which is indeed an inherent property of visual perception. The input data is encoded as tensors, then support information (including proximity and smoothness of continuity) is propagated by voting within a neighborhood.

For outlier removal, each point is represented by combining the location (x, y) and the motion (u, v) into a 4D tuple (x, y, u, v). Since each point is represented in the 4-dimensional space. A 4-D second order, symmetric, non-negative definite tensor is defined for each point, which is used to discover the outliers by examining the eigensystem.

First, each point is encoded as a ball tensor with eigenvalues and eigenvectors

$$\begin{aligned} &\lambda_1 = 1, \quad e_1 = (1, 0, 0, 0)^\top, \\ &\lambda_2 = 1, \quad e_2 = (0, 1, 0, 0)^\top, \\ &\lambda_3 = 1, \quad e_3 = (0, 0, 1, 0)^\top, \\ &\lambda_4 = 1, \quad e_4 = (0, 0, 0, 1)^\top. \end{aligned} \tag{2}$$

The second order, symmetric, non-negative definite tensor can be decomposed as

$$\begin{aligned} T &= \sum_{i=1}^{4} \lambda_i e_i e_i^\top \\ &= \sum_{i=1}^{4} [(\lambda_i - \lambda_{i+1}) \sum_{k=1}^{i} e_k e_k^\top] + \lambda_4 \sum_{i=1}^{4} e_i e_i^\top, \end{aligned} \tag{3}$$

where $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \lambda_4$ are the eigenvalues in descending order, and $e_1, \cdots, e_4$ are the corresponding eigenvectors. Note that eigenvectors are represented as column vectors.

In the voting stage, each point communicates information with its neighbors, and refines the information it carries, which is demonstrated in Fig. 3(a). Specifically, the votes that a receiver point P collects from its neighborhood is

$$\begin{aligned} R(P) &= \sum_{P_i \in \mathcal{N}(P)} W(P, P_i) \\ &= \sum_{P_i \in \mathcal{N}(P)} e^{-\frac{\|\vec{v}_i\|^2}{\sigma^2}} \left(I - \frac{\vec{v}_i \ \vec{v}_i^\top}{\|\vec{v}_i^\top \ \vec{v}_i\|} \right), \end{aligned} \tag{4}$$

where $\vec{v}_i = \overrightarrow{P_i P}$. The set of neighbors of P is denoted by $\mathcal{N}(P)$. $W(P, P_i)$ is the vote that the voter point P_i casts to the receiver point P. σ is the scale of voting that controls the range within which a voter can influence a receiver. Empirically, σ is set to 0.1, and it does not have a significant impact on the performance of the proposed method. A k-d tree is used to find neighborhood in 4D tuple quickly.

After voting, the mis-matched outliers of the interest points can be eliminated by examining the eigen-system of the tensors. The basic idea is that a tensor represents the structure of a manifold going through the point by encoding the normals to the manifold as eigenvectors of non-zero eigenvalues, and the tangents as eigenvectors of zero eigenvalues. Specifically, the saliency that a tensor only has d normals is defined by $\lambda_d - \lambda_{d+1}$. Therefore, the types of structures it encoded can be determined by the number of non-zero differences of consecutive eigenvalues.

In outlier removal, as illustrated in Fig. 3(b), correctly matched points are supposed to lie in a smooth surface in the 4D space, which have strong support, while the incorrectly matched points are likely to be isolated points in the 4D space, which receive little or no supports. Since a surface in 4D space is characterized by two normal vectors, the support of a point is measured by the *2D variety saliency*, which is defined by $\lambda_2 - \lambda_3$. Consequently, correctly matched

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2017.2763602, IEEE Transactions on Circuits and Systems for Video Technology

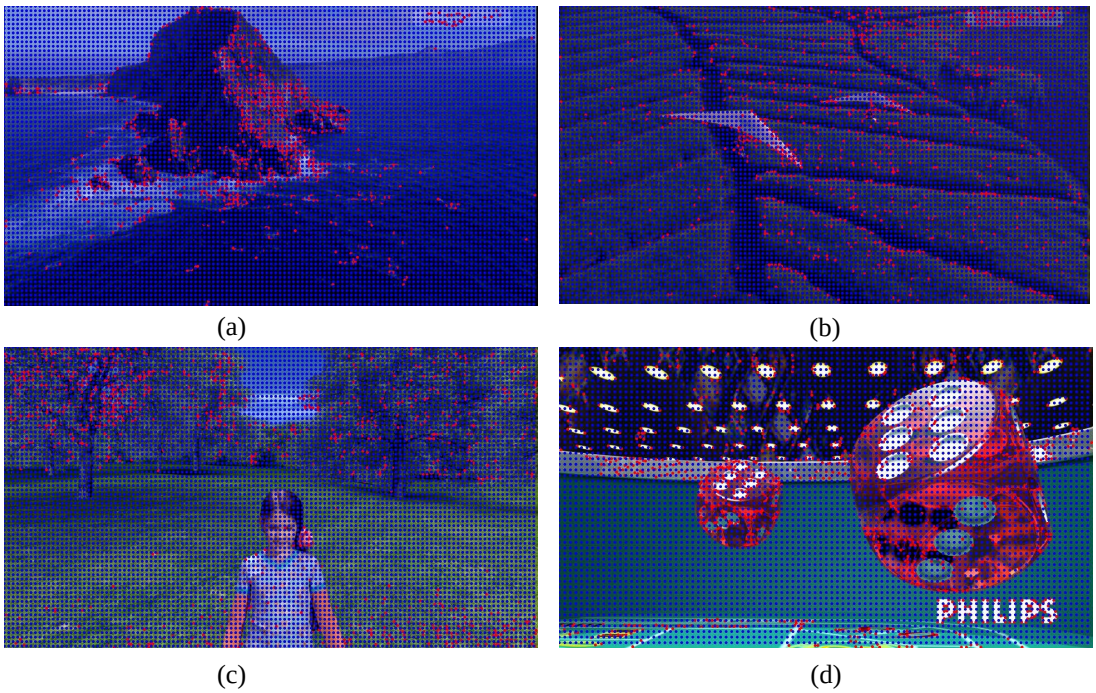


Fig. 2. Interest points sampled in testing sequences: red points indicates corners and blue points are uniformly sampled points. (Best viewed in color)

points are supposed to have large 2D variety saliency, while the incorrectly matched ones are supposed to have small 2D variety saliency, which is demonstrated in Fig. 3(c). Thus, the outliers can be removed by thresholding the 2D variety saliency of the points. In the experiments, the points with 2D variety saliency smaller than 0.75 of the average 2D variety saliency will be removed.

The algorithm of outlier removal by 4D tensor voting is summarized in Algorithm 1, and the sparse depth of the resulting reliable points can be attained by shifted bilateral filtering [6].

E. Depth Interpolation by High-Dimensional Tensor Voting

After the depth values of the initial points $\mathcal{P} = \{p_i\}_{i=1}^{N}$ are estimated as described in Section III-C and Section III-D, the depth of the remaining points in the depth map, which are denoted by $\mathcal{Q} = \{q_i\}_{i=1}^{M}$, will be attained by depth interpolation via 8D tensor voting.

Here, each point is represented by (x, y, u, v, r, g, b, d), where (x, y) and (u, v) are the location and displacement of the point, respectively. (r, g, b) are the values of the red, green and blue channels, and d is the depth of the point. In interpolation processing , we use the 8D tensor voting to find the relation $(x, y, u, v, r, g, b) \rightarrow d$. In order to estimate the depth for $\mathcal{Q}$, we take the 8D space as input-output space, where input space is (x, y, u, v, r, g, b) and output variable is d. Assuming that each point $P_i \in \mathcal{P}$ lies on a manifold, tensor voting can be used to extract local structures in this manifold.

After the data points are represented with tensors, they can refine the information they carry based on their neighbors through a voting process. The vote that the voter casts to the receiver has the orientation the receiver would have, if both the

Algorithm 1: Outlier Removal by 4D Tensor Voting

Input: N initial points: $\{P_i\}_{i=1}^{N}$, $P_i = (x_i, y_i, u_i, v_i)$
Output: Refined points
1. Initialization:
for $i = 1$ *to* N **do**
  Encode T_i as identity matrix I;
  Decompose T_i's eigensystem according to Eq. (3);
end
Construct k-d tree for fast neighbor searching;
2. Voting:
for $i = 1$ *to* N **do**
  for *each* $P_j \in \mathcal{N}(P_i)$ **do**
    Compute ball voting $W(P_j, P_i)$ according to Eq. (4);
    $R_i \leftarrow R_i + W(P_j, P_i)$;
  end
end
3. Analysis:
for $i = 1$ *to* N **do**
  Decompose R_i's eigensystem according to Eq. (3);
  Calculate $S_i \leftarrow \lambda_2 - \lambda_3$, $S_{avg} \leftarrow S_{avg} + S_i$;
end
Calculate $S_{avg} \leftarrow S_{avg}/N$;
Set $\alpha \leftarrow 0.75$;
for $i = 1$ *to* N **do**
  if $S_i < \alpha S_{avg}$ **then**
    Discard P_i;
  end
end

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2017.2763602, IEEE Transactions on Circuits and Systems for Video Technology

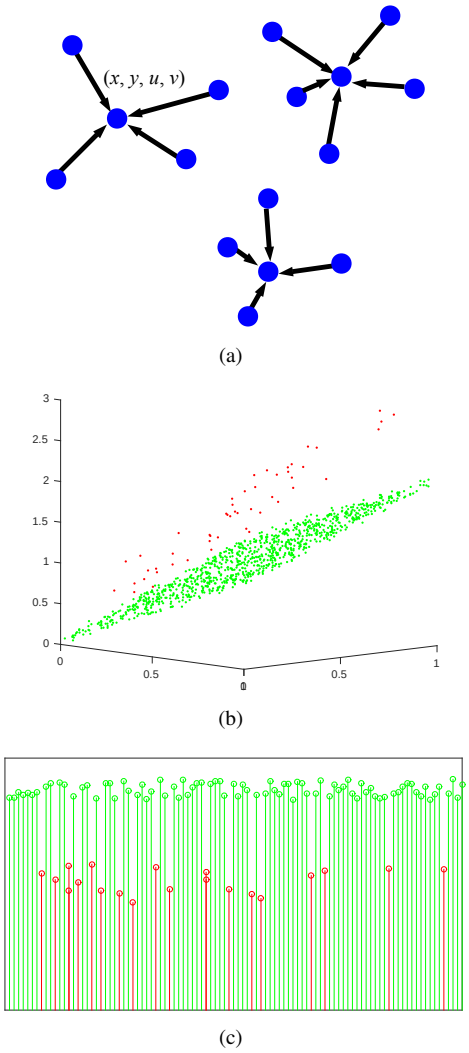


Fig. 3. Outlier removal by 4D tensor voting. (a) Ball voting; (b) visualization of tensors; (c) 2D variety saliency (i.e., $\lambda_2 - \lambda_3$). The green dots represent the correctly-matched points, which have strong support and high variety saliency. The red dots represent the falsely-matched points, which have weak support and low variety saliency.

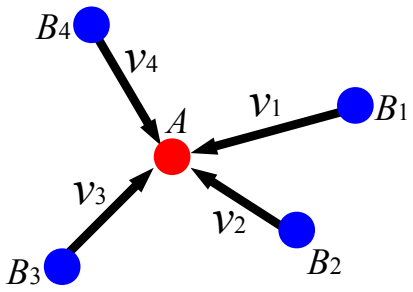


Fig. 4. Tensor voting process.

voter and receiver belong to the same structure. The magnitude of the vote is a function of the confidence we have that the voter and receiver indeed belong to the same structure. Since the data points in this paper are encoded as ball tensors, the process of ball voting will be introduced.

As illustrated in Fig. 4, the vote by a ball voter B_i propagates the voter's preference for a straight line that connects

TABLE II
TENSOR INTERPRETATION IN THE 8D SPACE

Dimension	Saliency	Normals	Tangents
0	λ_8	$e_1, e_2, \cdots, e_7, e_8$	none
1	$\lambda_7 - \lambda_8$	$e_1, e_2, \cdots, e_7$	e_8
2	$\lambda_6 - \lambda_7$	$e_1, e_2, e_3, e_4, e_5, e_6$	e_7, e_8
3	$\lambda_5 - \lambda_6$	e_1, e_2, e_3, e_4, e_5	e_6, e_7, e_8
4	$\lambda_4 - \lambda_5$	e_1, e_2, e_3, e_4	e_5, e_6, e_7, e_8
5	$\lambda_3 - \lambda_4$	e_1, e_2, e_3	e_4, e_5, e_6, e_7, e_8
6	$\lambda_2 - \lambda_3$	e_1, e_2	$e_3, e_4, \cdots, e_8$
7	$\lambda_1 - \lambda_2$	e_1	$e_2, e_3, \cdots, e_8$

Algorithm 2: Depth Interpolation by 8D Tensor Voting

Input: $\mathcal{P} = \{P_i\}_{i=1}^N$, $\mathcal{Q} = \{Q_i\}_{i=1}^M$
Output: The depth values of $\mathcal{Q}$: $\{d_i\}_{i=1}^M$
1. Initialization:
for $i = 1$ *to* N **do**
  Encode ball tensor T_i as identity matrix I for P_i;
end
Set P_t as the projection of P in input space;
Set Q_t as the projection of Q in input space;
Construct k-d trees of P and P_t for fast neighbor searching;
2. Tangent Space Calculation by Voting:
for $i = 1$ *to* N **do**
  Compute ball voting $R(P_i)$ according to Eq. (4);
end
for $i = 1$ *to* N **do**
  Decompose R_i's eigensystem according to Eq. (3);
  Calculate the tangent space V_i of P_i;
end
3. Depth Estimation for $\mathcal{Q}$:
for $i = 1$ *to* M **do**
  Find P' as the nearest neighbor of Q_i';
  Project $\overrightarrow{Q_i'P'}$ into V to get desirable direction $\vec{w}$;
  $\hat{Q} \leftarrow P' + \tau\vec{w}$;
  while $|\hat{Q} - Q_i'| > \epsilon$ **do**
    Set $\hat{Q}$ as a new start point;
    Calculate $\hat{Q}$'s tangent space and get desirable direction $\vec{w}$;
    $\hat{Q} \leftarrow \hat{Q} + \vec{w}$;
  end
  $d_i \leftarrow d_{\hat{Q}}$;
end

voter to the receiver A, which is the simplest and smoothest continuation between two points without other information provided. Hence, the vote of a ball voter is a tensor that spans the $(N-1)$-D normal space of the line and has one zero eigenvalue associated with the eigenvector that is parallel to the line. Meanwhile, the magnitude of the vote is dependent on the distance between the two points. Mathematically, the vote that the voter B_i casts to the receiver A in Fig. 4 is defined as

$$W(B_i, A) = e^{-\frac{\|v_i\|^2}{\sigma^2}} \left(I - \frac{v_i v_i^\top}{\|v_i v_i^\top\|} \right), \tag{5}$$

which is also a $N \times N$ tensor. In Eq. (5), v_i is a unit vector

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2017.2763602, IEEE Transactions on Circuits and Systems for Video Technology

parallel to $\overrightarrow{B_iA}$. Therefore, the votes that a receiver point collects from all of its neighbors are

$$\begin{aligned} R(A) &= \sum_{B_i \in \mathcal{N}(A)} W(A, B_i) \\ &= \sum_{B_i \in \mathcal{N}(A)} e^{-(\frac{\|v_i\|^2}{\sigma^2})} \left(I - \frac{\vec{v}_i \vec{v}_i^\top}{\|\vec{v}_i^\top \vec{v}_i\|} \right), \end{aligned} \tag{6}$$

where $\mathcal{N}(A)$ denotes the set of neighbors of point A, which is $\{B_1, B_2, B_3, B_4\}$ in Fig. 4.

The local structure is characterized by normal and tangent vectors. The structure in the 8D space can be represented as parametric equations: $x = x, y = y, r = r, g = g, b = b, u = u(x, y), v = v(x, y), d = d(x, y, r, g, b)$. Since these equations are controlled by 5 parameters, i.e., (x, y, r, g, b), the local structure can be characterized by 3 normal vectors and 5 tangent vectors, as shown in Table I. The 5 tangent vectors span a tangent space V_i of P_i, as Eq. (7)

$$V = span\{e_4, e_5, e_6, e_7, e_8\} \tag{7}$$

The local smoothness around P_i is kept in the derived tangent space. Thus, we can interpolate a new point in the neighborhood of P_i.

The algorithm of depth interpolation by 8D tensor voting is illustrated in Algorithm. 2. The 7D input space of point P is denoted by $P' = (x, y, u, v, r, g, b)$. To estimate the depth of a point $Q_i \in \mathcal{Q}$, we first find the nearest neighbor of Q_i', which is denoted by P'. Subsequently, the direction $\overrightarrow{P'Q_i'}$ is computed and project it back into the 8D space.

$$\vec{w} = V^\top (VV^\top)^{-1} V \overrightarrow{P'Q_i'} \tag{8}$$

The 8D point P is taken as the starting point on the manifold. The desired direction $\vec{w}$ is the projection of the vector $\overrightarrow{P'Q_i'}$ on the tangent space V of P. Then, we take a small step along $\vec{w}$ towards Q_i to get $\hat{Q}$, according to $\hat{Q} = P + \tau\vec{w}$. Approximation stops when $|\hat{Q} - Q_i| < \varepsilon$. $\hat{Q}$ in 8D space is the desired interpolated point for Q_i. Therefore, the depth value of Q_i is $d_{\hat{Q}}$.

IV. Experiments

The proposed method is evaluated upon the dataset used in [20], which is composed of ten sequences: *Inition-2d3d-Showreel-1*, *Inition-2d3d-Showreel-2*, *Philips-3D-experience-1*, *Philips-3D-experience-2*, *Dice-1*, *Dice-2*, *HeadRotate*, *Building*, *Interview* and *InnerGate*. The first eight sequences are collected from the Philips WowVc© project, and the last two sequences are from Heinrich-Hertz-Institute and [8], respectively. In particular, this dataset covers many challenging scenarios for depth estimation including textureless regions, occlusions, color ambiguity and fast-moving objects. A comprehensive summarization of the dataset is illustrated in Table III.

A. Comparison with State-of-the-art Methods

In the first experiment, we evaluate the performance of the proposed method both objectively and subjectively in comparison with many state-of-the-art methods, including

- Bilateral filtering (BF) [24];
- Improved depth propagation using keyframes (IDP) [4];
- Disparity propagation (DP) [6];
- Bi-directional motion estimation (BDME) [8];
- Natural scene statistics models (NSS) [20];
- Example-based video stereolization (EBVS) [7].

The mean squared error (MSE) of the estimated depth maps by the seven methods is displayed in Table IV. In general, the proposed method achieves the lowest MSE in eight out of ten test sequences. In the *Philips-3D-experience-2* sequence, although the proposed method achieves the second best result among the seven method, it is outperformed by NSS by large margin, because natural scene statistics provides a robust prior guide in modeling the natural scenes.

In addition to MSE, we also evaluate the Structure Similarity (SSIM) indices of the seven methods, because this metric is more characteristic of the structural information that the viewers are particularly sensitive to. The SSIM reflects the similarity of two depth maps in structure, and is implemented in the LIVE website [25]. The results of SSIM evaluation are displayed in Table V. Again, the proposed method achieves the best performance among the seven methods, which obtains the highest SSIM in eight out of ten sequences. Table V clearly demonstrates that the proposed method is more effective in preserving the structure information and the discontinuities in depth estimation.

Furthermore, several examples of the estimated depth maps by IDP, BDME, EBVS and the proposed method are displayed in Fig. 5, and the ground truth depth maps are shown in the last column in Fig. 5. In particular, we enlarge some local patches lying in the depth boundary, which are displayed below the depth maps for each sequence. Moreover, the PSNRs of the estimated depth maps are also displayed below the frames. It is clearly shown that the proposed method can obtain reliable depth estimation in occlusion boundary areas. In *Philips-3D-experience-1*, due to occlusion, EBVS misuses the depth of moving foreground object to estimate that of occluded background areas, while the proposed method accurately captures the depth discontinuities of the foreground girl and the background lawn.

B. Error Accumulation Across Frames

In addition to the global evaluation of depth estimation, we further plot the MSE of each frame estimated by IDP, BDME, EBVS and the proposed method for the *Philips-3D-experience-1*, *Dice-1*, *HeadRotate* and *Interview* sequences.

A key-frame with ground truth depth map is provided every 20 frames for each sequence, from which the depth maps of other frames are estimated. To guarantee the accuracy, the depth map of a non key-frame is linearly interpolated from the two depth maps propagated from its previous key-frame and next key-frame (except for IDP, which propagates depth causally).

The MSE curves are illustrated in Fig. 6. We can observe from Fig. 6 that the MSE curves oscillate periodically at an interval of 20, and peak in the middle of the key-frames for most test sequences. This is because the frames in the middle

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2017.2763602, IEEE Transactions on Circuits and Systems for Video Technology

TABLE III
TESTING SEQUENCE SUMMARIZATION

Sequence	Resolution	Frames	Key-frame interval	Compositions
Inition-2d3d-Showreel-1	960 × 540	91	30	outdoor scene, textureless, zoom in/out
Inition-2d3d-Showreel-2	960 × 540	91	30	outdoor scene, large displacement, sharp edges
Philips-3D-experience-1	960 × 540	41	20	outdoor scene, occlusion, sharp edges
Philips-3D-experience-2	960 × 540	61	30	outdoor scene, thin objects
Dice-1	960 × 540	61	20	indoor scene, textureless, sharp edges
Dice-2	960 × 540	101	25	indoor scene, occlusion, sharp edges
HeadRotate	960 × 528	81	20	indoor scene, occlusion, color ambiguity
Building	360 × 184	101	25	indoor scene, sharp edges, color ambiguity
Interview	720 × 184	101	25	indoor scene, occlusion,large displacement
InnerGate	640 × 384	901	25	outdoor scene, large displacement, zoom in/out

TABLE IV
MEAN SQUARED ERROR OF DEPTH ESTIMATION

	BF [24]	IDP [4]	DP [6]	BDME [8]	NSS [20]	EBVS [7]	Proposed
Inition-2d3d-Showreel-1	42.59	40.91	47.46	16.89	**10.13**	32.68	11.28
Inition-2d3d-Showreel-2	7.76	7.55	8.51	5.51	4.87	6.32	**3.97**
Philips-3D-experience-1	87.13	94.83	40.04	41.98	26.87	32.68	**25.12**
Philips-3D-experience-2	607.15	548.94	245.50	190.77	**102.34**	388.20	163.23
Dice-1	131.40	124.75	249.98	86.97	55.10	113.80	**52.79**
Dice-2	71.18	70.01	191.54	69.25	40.55	79.88	**37.92**
HeadRotate	85.70	79.78	40.58	19.27	17.09	57.99	**16.20**
Building	387.67	360.47	227.29	105.81	84.01	192.33	**69.28**
Interview	112.32	98.73	68.73	45.03	31.76	68.32	**23.13**
InnerGate	497.47	529.96	400.77	156.41	114.52	195.30	**98.55**

TABLE V
STRUCTURE SIMILARITY COMPARISON

	BF [24]	IDP [4]	DP [6]	BDME [8]	NSS [20]	EBVS [7]	Proposed
Inition-2d3d-Showreel-1	.967	.971	.974	.979	**.982**	.973	.980
Inition-2d3d-Showreel-2	.984	.985	.985	.981	.985	.983	**.991**
Philips-3D-experience-1	.961	.971	.977	.976	**.980**	.975	**.980**
Philips-3D-experience-2	.912	.928	.933	.935	.947	.935	**.967**
Dice-1	.978	.988	.978	.985	.988	.982	**.989**
Dice-2	.983	.990	.981	.987	**.991**	.990	.99
HeadRotate	.973	.976	.981	.987	.988	.979	**.990**
Building	.840	.875	.902	.922	.928	.912	**.932**
Interview	.951	.963	.976	.979	**.984**	.979	**.984**
InnerGate	.891	.900	.913	.930	.937	.917	**.949**

TABLE VI
PERFORMANCE OF DIFFERENT OUTLIER REMOVAL STRATEGIES

	without outlier removal	Optical flow validation	4D tensor voting
Inition-2d3d-Showreel-1	32.19	16.91	**11.28**
Inition-2d3d-Showreel-2	21.61	9.15	**3.97**
Philips-3D-experience-1	77.36	31.35	**25.12**
Philips-3D-experience-2	401.14	268.14	**163.23**
Dice-1	194.04	124.57	**52.79**
Dice-2	105.82	90.31	**37.92**
HeadRotate	41.13	24.18	**16.20**
Building	80.17	76.21	**69.28**
Interview	109.12	28.30	**23.13**
InnerGate	419.17	186.16	**98.55**

of two key-frames have relatively large error accumulation from both sides, whereas the frames near the key-frames have smaller error from the closer key-frames suppressed by linear interpolation. A special case is Fig. 6 (a), where the error peaks are closer to the future key-frames, because it is likely that for this video the forward propagation is more accurate than backward propagation.

C. Performance of Outlier Removal by 4D Tensor Voting

Furthermore, to evaluate effectiveness of the proposed outlier removal algorithm via 4D tensor voting, we compare it with other two strategies, namely, without outlier removal and bi-directional optical flow validation [20]. In particular, for bi-directional optical flow validation, the forward and backward motion vectors will be computed for each interest point, which are denoted by $\boldsymbol{v}_f$ and $\boldsymbol{v}_b$. In the ideal case, these two motion vectors should be opposite, i.e., $\|\boldsymbol{v}_f + \boldsymbol{v}_b\|^2 = 0$. Hence, an

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2017.2763602, IEEE Transactions on Circuits and Systems for Video Technology

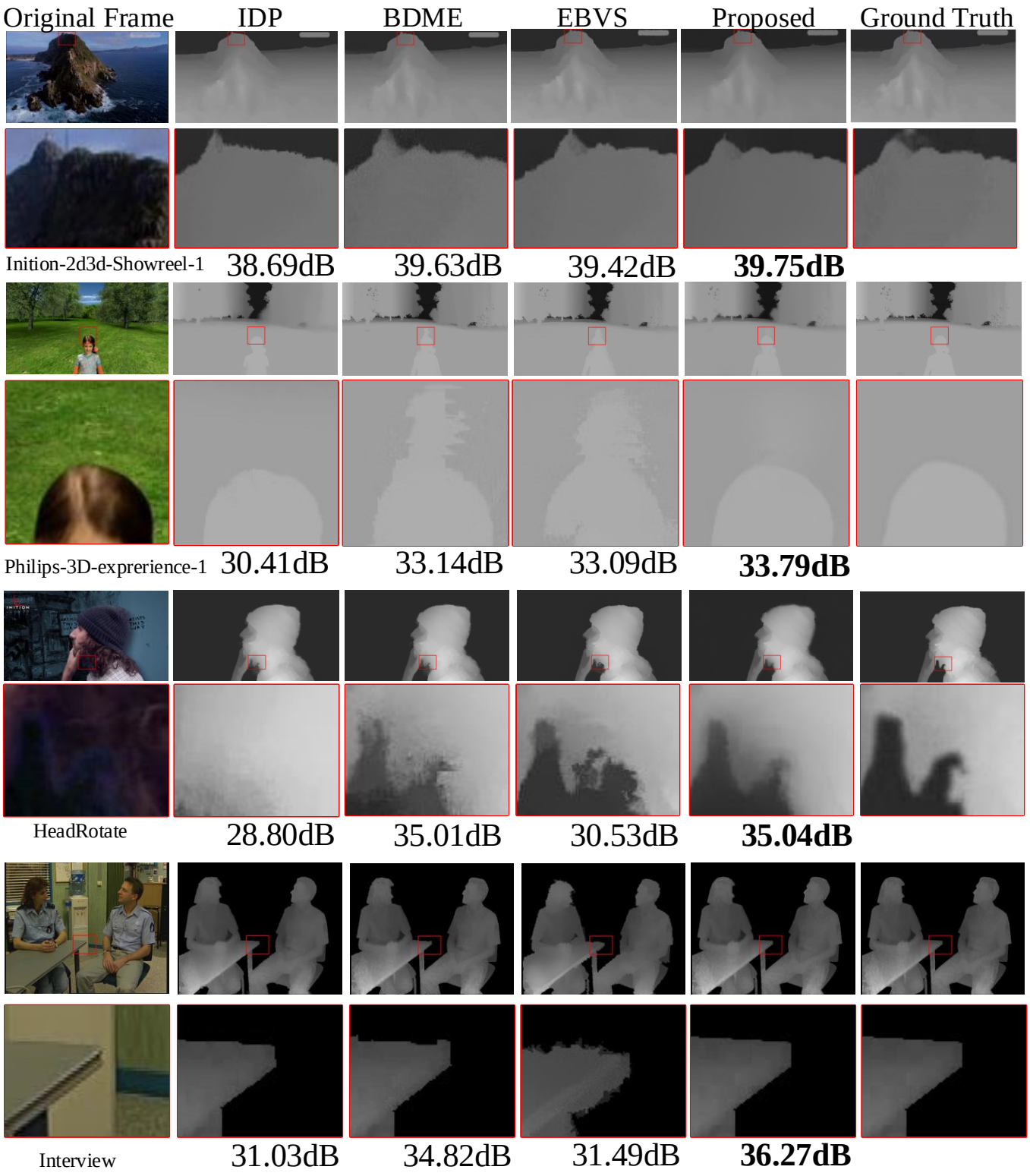
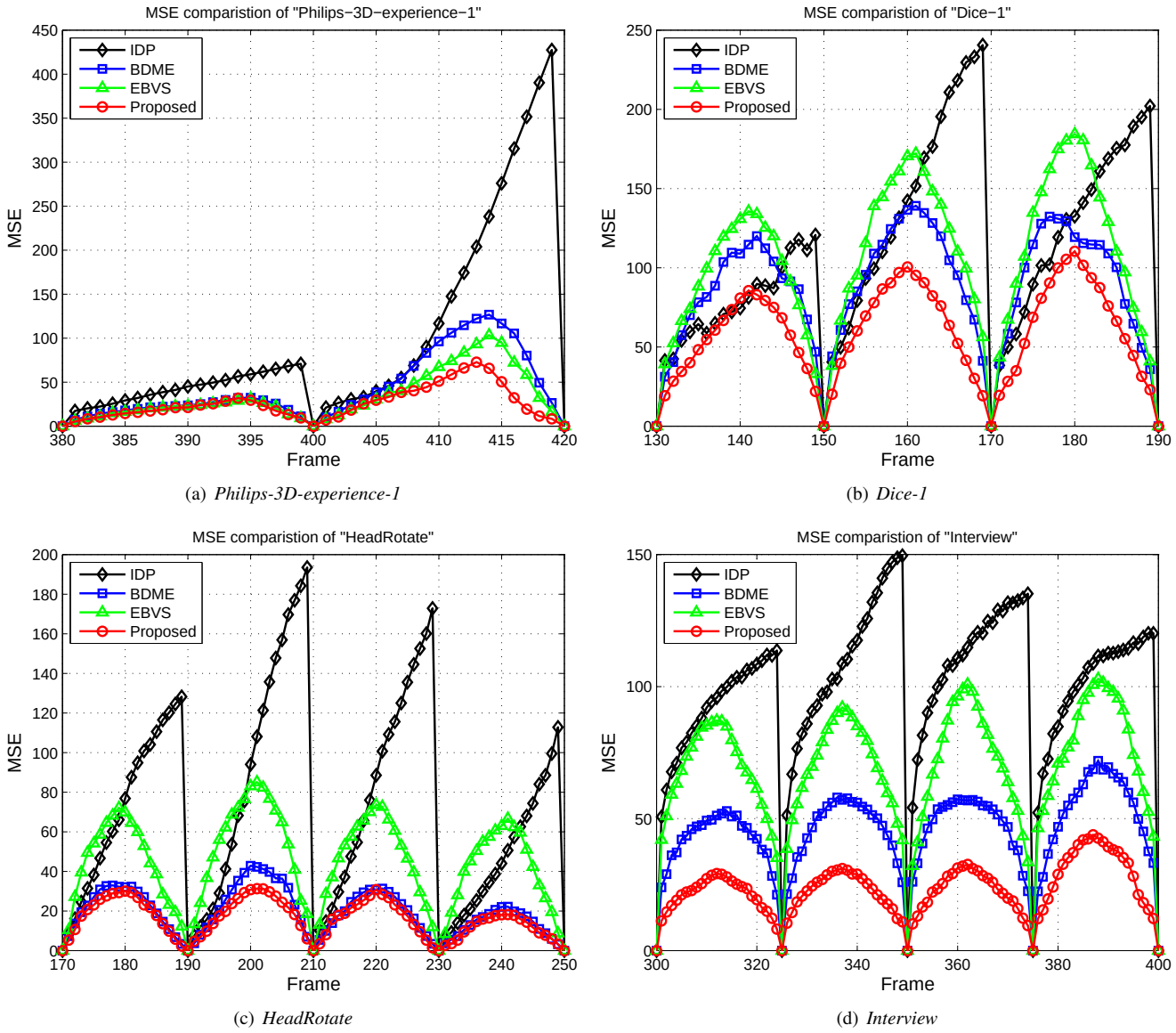


Fig. 5. Estimated depth maps of four testing sequences (from top to bottom): *Inition-1*, *Philips-1*, *HeadRotate* and *Interview*. The PSNR values of depth maps are shown below the frames.

interest point will be regarded as an outlier if $\|\boldsymbol{v}_f + \boldsymbol{v}_b\|^2 > 2$. The performance of the three outlier removal strategies is displayed in Table VI.

As expected, using all the interest points without removing the outliers brings large error to the depth map estimation due to inaccurate motion estimation, occlusion and deformation. On the other hand, the error can be reduced by validating the forward and backward optical flow vectors, because mismatched interest points are discarded, whose forward and backward motion vectors have large deviations. Finally, the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2017.2763602, IEEE Transactions on Circuits and Systems for Video Technology



(a) *Philips-3D-experience-1* (b) *Dice-1*

(c) *HeadRotate* (d) *Interview*

Fig. 6. Frame-wise mean squared error by IDP, BDME, EBVS and the proposed method.

TABLE VII
AVERAGE TIME CONSUMPTION FOR EACH STEP OF THE PROPOSED METHOD (IN SECONDS)

	Point extraction	KLT	Outlier removal	Sparse depth	Depth interpolation	Total
Inition-2d3d-Showreel-1	0.04	0.32	8.16	0.11	53.13	61.92
Inition-2d3d-Showreel-2	0.04	0.31	7.98	0.11	57.32	66.03
Philips-3D-experience-1	0.04	0.35	8.26	0.11	50.42	59.39
Philips-3D-experience-2	0.04	0.35	8.23	0.11	60.13	69.10
Dice-1	0.04	0.30	6.91	0.11	52.22	59.75
Dice-2	0.04	0.30	6.01	0.11	51.62	58.32
HeadRotate	0.04	0.33	7.38	0.10	54.41	62.41
Building	0.02	0.11	1.69	0.03	13.12	15.18
Interview	0.03	0.27	4.46	0.09	48.47	53.49
InnerGate	0.03	0.28	5.33	0.08	49.08	55.05

best performance is achieved by the proposed 4D tensor voting method, because it is capable of preserving the structural information, which is critical in modeling the discontinuities in the depth field.

D. Voting Neighborhood

We evaluate the influence of voting neighborhood to the performance of proposed depth interpolation on *Inition-2d3d-Showreel-1*, *Philips-3D-experience-1*, *HeadRotate* and *Interview*. We change the voting neighborhood from 10 to 100 by

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2017.2763602, IEEE Transactions on Circuits and Systems for Video Technology

stride of 5, and the relation of MSE with neighborhood size is demonstrated in Fig. 7.

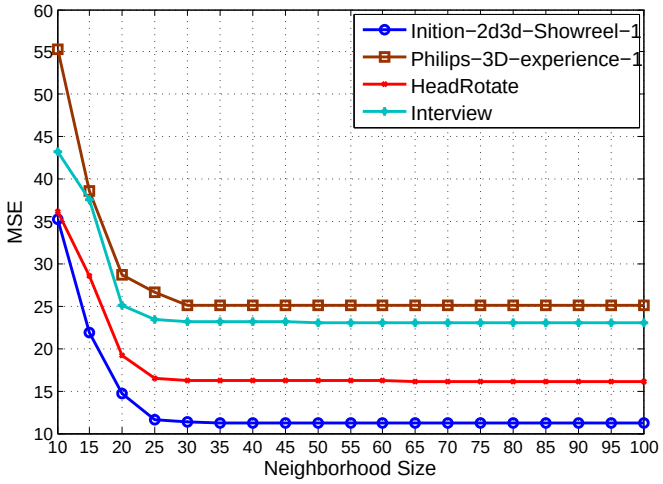


Fig. 7. Relation of MSE and neighborhood size.

It can be observed from Fig. 7 that MSE drops drastically as neighborhood size increases before 25. When the neighborhood size exceeds 25, the MSE almost converges, since the voting weights are trivial from distant points.

E. Computational Complexity

Finally, we present the computational complexity of the proposed method. The experiments are conducted on a PC with 3.2GHz Intel Core-i5 CPU and 8GB RAM. The algorithm is implemented in C++ and OpenCV library. The average time consumption of interest point extraction, KLT tracking, outlier removal, sparse depth computation and depth interpolation by high-dimensional tensor voting for each testing sequence is illustrated in Table VII.

In general, the proposed method takes about one minute to estimate the depth map for each frame. Most of the run-time is spent on depth interpolation, since there are large quantity of points to be inferred, and the tensor voting process is more complicated in the 8D space.

F. Analysis

From the experiments, we can observe that the proposed method achieves accurate depth estimation, and outperforms other methods both qualitatively and quantitatively. The reason why tensor voting makes such improvement for depth estimation is two-fold.

First, since the sparse depth field is critical for the proposed method, tensor voting is utilized to remove false matches of interest points, thus, generating reliable depth vectors as the initialization of the following depth propagation. Compared with other outlier removal approaches, 4D tensor voting achieves better result, because it investigates the eigen-system of the matching points, which have strong support for correctly matched ones and weak support for falsely matched ones in sense of texture and motion.

Second, based on this reliable initial depth field, high dimensional tensor voting is used to propagate the depth from the interest points to the entire image. Compared with conventional methods that propagate depth homogeneously, the proposed method propagates depth using high dimensional tensor voting in a structure-aware fashion. Hence, a point would receive strong directed votes from points that lie in the same structure and less from the others, making the depth consistent with the motion and texture of the frames.

V. Conclusions

This paper presents a structure-aware sparse-to-dense depth estimation algorithm, which leverages tensor voting for outlier removal as well as depth propagation. On the one hand, a 4D tensor voting algorithm is designed to eliminate outlier points with inaccurate motion estimation. Hence, the depth of a sparse set of high-confidence interest points can be obtained, which will be propagated to the other region of the image. On the other hand, a high-dimensional tensor voting algorithm, incorporating spatial location, motion and color into the tensor representation, is devised to propagate the depth from the sparse points to the entire image domain. Experimental result shows that the proposed method outperforms many state-of-the-art depth estimation approaches on public benchmarks.

References

[1] C. Fehn, "Depth-image-based rendering (DIBR), compression, and transmission for a new approach on 3D-TV," in *Proc. SPIE, Stereoscopic Displays and Virtual Reality Systems XI*, San Jose, CA, USA, May 2004, pp. 93–104. 1

[2] Y. Zhao, C. Zhu, Z. Chen, and L. Yu, "Depth no-synthesis-error model for view synthesis in 3-d video," *IEEE Tran. Image Processing*, vol. 20, no. 8, pp. 2221–2228, Aug. 2011. 1

[3] M. Kim, S. Park, H. Kim, and I. Artem, "Automatic conversion of two-dimensional video into stereoscopic video," in *Proc. SPIE, Three-Dimensional TV, Video, and Display IV*, Boston, MA, USA, Nov. 2005, pp. 601–610. 1

[4] C. Varekamp and B. Barenbrug, "Improved depth propagation for 2D-to-3D video conversion using key-frames," in *Proc. European Conf. Visual Media Production*, London, UK, Nov. 2007, pp. 1–7. 1, 3, 7, 8

[5] M. Guttmann, L. Wolf, and D. Cohen-Or, "Semi-automatic stereo extraction from video footage," in *Proc. Int'l Conf. Computer Vision (ICCV'09)*, Kyoto, Japan, Sept. 2009, pp. 136–142. 1, 2

[6] X. Cao, Z. Li, and Q. Dai, "Semi-automatic 2D-to-3D conversion using disparity propagation," *IEEE Trans. Broadcast.*, vol. 57, no. 2, pp. 491–499, June 2011. 1, 2, 3, 5, 7, 8

[7] L. Wang and C. Jung, "Example-based video stereolization with foreground segmentation and depth propagation," *IEEE Trans. Multimedia*, vol. 16, no. 7, pp. 1905–1914, Nov. 2014. 1, 2, 3, 7, 8

[8] Z. Li, X. Cao, and Q. Dai, "A novel method for 2D-to-3D video conversion using bi-directional motion estimation," in *Proc. Int'l Conf. Acoustics, Speech and Signal Processing (ICASSP'12)*, Kyoto, Japan, Mar. 2012, pp. 1429–1432. 1, 3, 7, 8

[9] R. Moreno, M. Garcia, D. Puig, L. Pizarro, B. Burgeth, and J. Weickert, "On improving the efficiency of tensor voting," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 33, no. 11, pp. 2215–2228, Nov. 2011. 1

[10] T. Wu, S. Yeung, J. Jia, C. Tang, and G. Medioni, "A closed-form solution to tensor voting: Theory and applications," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 34, no. 8, pp. 1482–1495, Aug. 2012. 1, 3

[11] R. Zhang, P. Tsai, J. Cryer, and M. Shah, "Shape-from-shading: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 8, pp. 690–706, Aug. 1999. 2

[12] S. Bhasin and S. Chaudhuri, "Depth from defocus in presence of partial self occlusion," in *Proc. Int'l Conf. Computer Vision (ICCV'01)*, Vancouver, BC, Canada, Jul. 2001, pp. 1623–1630. 2

[13] X. Lin, J. Suo, and Q. Dai, "Extracting depth and radiance from a defocused video pair," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 25, no. 4, pp. 557–569, April 2015. 2

[14] Z. Zhang, C. Zhou, Y. Wang, and W. Gao, "Interactive stereoscopic video conversion," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 23, no. 10, pp. 1795–1808, Oct. 2013. 2

[15] K. Moustakas, D. Tzovaras, and M. Strintzis, "Stereoscopic video generation based on efficient layered structure and motion estimation from a monoscopic image sequence," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 15, no. 8, pp. 1065–1073, Aug. 2005. 2

[16] D. Crandall, A. Owens, N. Snavely, and D. Huttenlocher, "Sfm with mrfs: Discrete-continuous optimization for large-scale structure from motion," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 35, no. 12, pp. 2841–2853, Dec. 2013. 2

[17] J. Lei, S. Li, C. Zhu, M. T. Sun, and C. Hou, "Depth coding based on depth-texture motion and structure similarities," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 25, no. 2, pp. 275–286, Feb. 2015. 2

[18] J. Konrad, M. Wang, P. Ishwar, C. Wu, and D. Mukherjee, "Learning-based, automatic 2D-to-3D image and video conversion," *IEEE Trans. Image Processing*, vol. 22, no. 9, pp. 3485–3496, Sept. 2013. 2

[19] T. Wu, S. Yeung, J. Jia, and C. Tang, "Quasi-dense 3d reconstruction using tensor-based multiview stereo," in *IEEE Conf. Computer Vision and Pattern Recognition (CVPR'10)*, San Francisco, CA, USA, June 2010, pp. 13–18. 2

[20] W. Huang, X. Cao, K. Lu, Q. Dai, and A. Bovik, "Toward naturalistic 2D-to-3D conversion," *IEEE Trans. Image Process.*, vol. 24, no. 2, pp. 724–733, Feb. 2015. 3, 7, 8

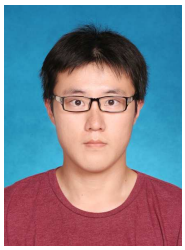
[21] G. Medioni, M. Lee, and C. Tang, "A computational framework for segmentation and grouping," *Elsevier Science*, 2000. 3

[22] J. Shi and C. Tomasi, "Good features to track," in *IEEE Conf. Computer Vision and Pattern Recognition (CVPR'94)*, Seattle, WA, USA, June 1994, pp. 593–600. 3, 4

[23] G. Medioni, C. Tang, and M. Lee, "Tensor voting: Theory and applications," in *Proc. RFIA*, Paris, France, 2000. 4

[24] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proc. Int'l Conf. Computer Vision (ICCV'98)*, Bombay, India, Mar. 1998, pp. 839–846. 7, 8

[25] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, "The SSIM index for image quality assessment," http://www.cns.nyu.edu/~lcv/ssim, 2003. 7

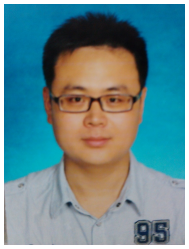


Botao Wang received the B.S. and Ph.D degrees from Shanghai Jiao Tong University, Shanghai, China, in 2010 and 2016, all in electronic engineering. His research interests include object detection, scene classification, and image understanding.

Junni Zou (M'07) received the M.S. degree and the Ph.D. degree in communication and information system from Shanghai University, Shanghai, China, in 2004 and 2006, respectively. Currently, she is a full Professor with the Department of Computer Science and Engineering, Shanghai Jiao Tong University (SJTU), China. From 2006 to 2016, she was with the School of Communication and Information Engineering, Shanghai University, Shanghai, and became a full Professor in 2014. From June 2011 to June 2012, she was with the Department of Electrical and Computer Engineering, University of California, San Diego (UCSD), as a visiting Professor.

Her research interests include multimedia communication, network resource optimization, wireless communication, and network information theory. She has published over 80 IEEE journal/conference papers, and 2 book chapters, including 16 IEEE Transactions journal papers. She holds 12 patents and has 10+ under reviewing patents.

Dr. Zou was granted National Science Fund for Outstanding Young Scholar in 2016. She was a recipient of Shanghai Yong Rising Star Scientist award in 2011. She obtained the First Prize of the Shanghai Technological Innovation Award in 2011, and the First Prize of the Shanghai Science and Technology Advancement Award in 2008. Also, she has served on some technical program committees for the IEEE and other international conferences.



Yong Li received the B.S. degree from Xuzhou Normal University, and the M.S. degree from China University of Mining and Technology, Xuzhou, China, in 2009 and 2012. He is currently pursuing the Ph.D. degree with the Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai, China. From Nov. 2014 to May 2015, Nov. 2015 to May 2016, he was with the Department of Biomedical Informatics, University of California, San Diego (UCSD), as a visiting scholar.

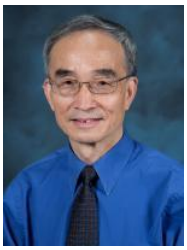
His current research interests include compressive sensing, sparse representation and image/signal processing.

Kuanyu Ju received the B.S. and M.S. degrees in electronic engineering from Shanghai Jiao Tong University, Shanghai, China, in 2013 and 2016, respectively.

Hongkai Xiong (M'01-SM'10) received the Ph.D. degree in communication and information system from Shanghai Jiao Tong University (SJTU), Shanghai, China, in 2003. Since then, he has been with the Department of Electronic Engineering, SJTU, where he is currently a full Professor. From December 2007 to December 2008, he was with the Department of Electrical and Computer Engineering, Carnegie Mellon University (CMU), Pittsburgh, PA, USA, as a Research Scholar. From 2011 to 2012, he was a Scientist with the Division of Biomedical Informatics at the University of California (UCSD), San Diego, CA, USA.

His research interests include source coding/network information theory, signal processing, computer vision and machine learning. He has published over 190 refereed journal/conference papers. He was the recipient of the Top 10% Paper Award at the 2016 IEEE Visual Communication and Image Processing (IEEE VCIP'16), the Best Student Paper Award at the 2014 IEEE Visual Communication and Image Processing (IEEE VCIP'14), the Best Paper Award at the 2013 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (IEEE BMSB'13), and the Top 10% Paper Award at the 2011 IEEE International Workshop on Multimedia Signal Processing (IEEE MMSP'11).

In 2016, Dr. XIONG was granted the Yangtze River Scholar Distinguished Professor from Ministry of Education of China, and the Youth Science and Technology Innovation Leader from Ministry of Science and Technology of China. He was awarded Shanghai Academic Research Leader. In 2014, he was granted National Science Fund for Distinguished Young Scholar and Shanghai Youth Science and Technology Talent as well. In 2013, he was awarded a recipient of Shanghai Shu Guang Scholar. From 2012, he is a member of Innovative Research Groups of the National Natural Science. In 2011, he obtained the First Prize of the Shanghai Technological Innovation Award for "Network-oriented Video Processing and Dissemination: Theory and Technology". In 2010 and 2013, he obtained the SMC-A Excellent Young Faculty Award of Shanghai Jiao Tong University. In 2009, he was awarded a recipient of New Century Excellent Talents in University, Ministry of Education of China. He served as TPC members for prestigious conferences such as ACM Multimedia, ICIP, ICME, and ISCAS. He is a senior member of the IEEE (2010).



Yuan F. Zheng (F'97) received the MS and Ph.D. degrees in Electrical Engineering from The Ohio State University, in Columbus, Ohio in 1980 and 1984, respectively. His undergraduate education was received at Tsinghua University, Beijing, China in 1970. From 1984 to 1989, he was with the Department of Electrical and Computer Engineering at Clemson University, Clemson, South Carolina. Since August 1989, he has been with The Ohio State University, where he is currently Winbigler Designated Chair Professor and was the Chairman of the Department of Electrical and Computer Engineering from 1993 to 2004. From 2004 to 2005, Professor Zheng spent sabbatical year at the Shanghai Jiao Tong University in Shanghai, China and continued to be involved as Dean of School of Electronic, Information and Electrical Engineering until 2008. Professor Zheng is an IEEE Fellow.

Professor Zheng's research interests include two aspects. One is in wavelet transform for image and video, radar waveform and signal, and object tracking. The other is in robotics which includes robotics for life science applications, multiple robots coordination, legged walking robots, and service robots. Professor Zheng has been on the editorial board of five international journals. For his research contributions, Professor Zheng received the Presidential Young Investigator Award from President Ronald Reagan in 1986, and the Research Awards from the College of Engineering of The Ohio State University in 1993, 1997, and 2007, respectively. Professor Zheng along with his students received the best conference and best student paper award a few times in 2000, 2002, 2006, 2009, and 2010, respectively, and received the Fred Diamond for Best Technical Paper Award from the Air Force Research Laboratory in Rome, New York in 2006. In 2004 and 2005, Professor Zheng was appointed to the International Robotics Assessment Panel by the NSF, NASA, and NIH to assess the robotics technologies worldwide. Most recently in 2017 he received the Innovator of the Year Award from The Ohio State University.