Zhao XR, Wang X, Chen QC. Temporally consistent depth map prediction using deep convolutional neural network and spatial-temporal conditional random field. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 32(3): 443–456 May 2017. DOI 10.1007/s11390-017-1735-x

Temporally Consistent Depth Map Prediction Using Deep Convolutional Neural Network and Spatial-Temporal Conditional Random Field

Xu-Ran Zhao, Xun Wang *, *Senior Member, CCF, Member, ACM, IEEE*, and Qi-Chao Chen

School of Computer and Information Engineering, Zhejiang Gongshang University, Hangzhou 310018, China

E-mail: {zxr,wx,14060401006}@zjgsu.edu.cn

Received December 23, 2016; revised March 20, 2017.

Abstract Deep convolutional neural networks (DCNNs) based methods recently keep setting new records on the tasks of predicting depth maps from monocular images. When dealing with video-based applications such as 2D (2-dimensional) to 3D (3-dimensional) video conversion, however, these approaches tend to produce temporally inconsistent depth maps, since their CNN models are optimized over single frames. In this paper, we address this problem by introducing a novel spatial-temporal conditional random fields (CRF) model into the DCNN architecture, which is able to enforce temporal consistency between depth map estimations over consecutive video frames. In our approach, temporally consistent superpixel (TSP) is first applied to an image sequence to establish the correspondence of targets in consecutive frames. A DCNN is then used to regress the depth value of each temporal superpixel, followed by a spatial-temporal CRF layer to model the relationship of the estimated depths in both spatial and temporal domains. The parameters in both DCNN and CRF models are jointly optimized with back propagation. Experimental results show that our approach not only is able to significantly enhance the temporal consistency of estimated depth maps over existing single-frame-based approaches, but also improves the depth estimation accuracy in terms of various evaluation metrics.

Keywords depth estimation, temporal consistency, convolutional neural network, conditional random fields

1 Introduction

Predicting depth maps from video sequences is a fundamentally important and challenging problem in computer vision. Depth usually provides valuable information, and facilitates applications in various fields, including 3D (3-dimensional) modeling[1], pose recognition[2], image-based rendering[3] and human computer interaction[4]. Moreover, in the growing 3D movie industry, knowing the depth information greatly simplifies the process of converting 2D (2-dimensional) movies to their stereoscopic form[5]. While depth estimation methods from images or video sequences have been extensively studied, most of them rely on specific depth cues such as camera motion[6], scene geometry[7] and shading[8]. These approaches make highly restrictive assumptions on scene types and can only find quality depth estimation for special cases. On the other hand, depth estimation approaches based on machine learning have gained more and more popularity because of their independence from scene-specific cues. In particular, Eigen *et al.*[9] first introduced the deep learning models based on convolutional neural networks (CNNs) into the depth estimation task and improved the estimation accuracy by over 30% compared with traditional methods. After that, the research on CNN-based depth estimation

Regular Paper

Special Section of CVM 2017

This work is supported in part by the Natural Science Foundation of Zhejiang Province of China under Grant No. LQ17F030001, the National Natural Science Foundation of China under Grant No. U1609215, Qianjiang Talent Program of Zhejiang Province of China under Grant No. QJD1602021, the National Key Technology Research and Development Program of the Ministry of Science and Technology of China under Grant No. 2014BAK14B01, and Beihang University Virtual Reality Technology and System National Key Laboratory Open Project under Grant No. BUAA-VR-16KF-17.

*Corresponding Author

proliferates, either building deeper and more complex CNN models[10], or exploiting spatial relationships of neighbouring (super)pixels[11-12] to improve depth estimation accuracy. However, all these methods operate on single 2D images while ignoring temporal coherence. When applied to continuous image sequences, these approaches tend to produce temporally inconsistent depth maps: the depth estimation of the same object jumps obviously in two consecutive frames. For example, Fig.1 shows the depth map estimations for three consecutive video frames of a bedroom scene using the recently proposed deep convolutional neural fields (DCNF) model[13]. Severe inter-frame saltations in depth estimation at several regions are noticed, with relative differences up to 30% of the ground truth depths. This effect causes serious problems in applications such as automatic 3D video generation because the temporal inconsistency in depth maps will propagate to synthesized 3D videos and can be easily perceived by human eyes.

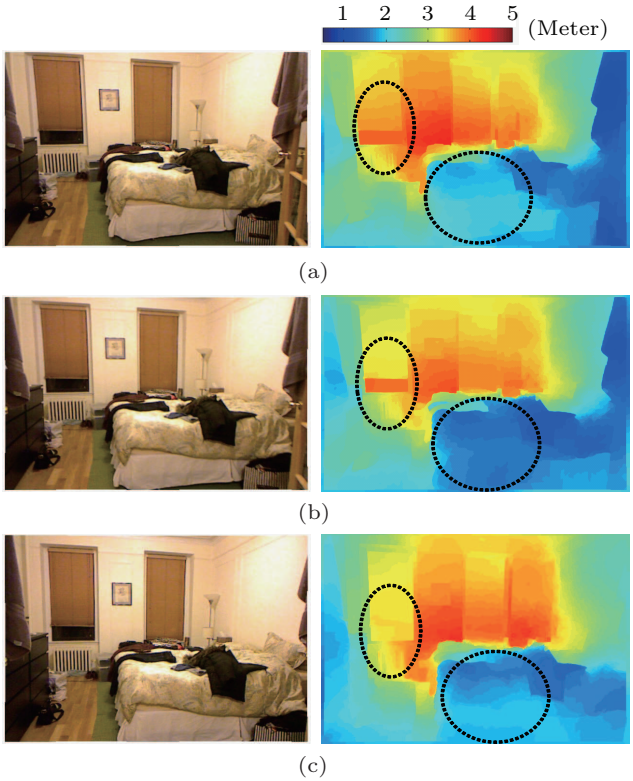


Fig.1. Estimated depth maps for 3 consecutive frames of a video using the deep convolutional neural fields (DCNF) model[13]. Temporal discontinuities of estimated depth are spotted at several regions, as shown in dashed ellipses. (a) Frame $t-1$. (b) Frame t. (c) Frame $t+1$.

In this paper, we extend single-image based CNN depth estimation to video-based applications by jointly optimizing the depth estimation of several consecutive frames and enforcing the temporal consistency of estimated depth maps between them. More specifically, we formulate video-based depth estimation as a deep structured regression problem: a CNN network is trained to predict a depth value for each single superpixel of the input video, and a conditional random fields (CRF) model is used to explicitly model the depth predictions of neighbouring superpixels in one frame and corresponding superpixels between two consecutive frames. Our work can be seen as an extension to the DCNF model proposed in [12] by adding a temporal dimension: not only are neighbouring superpixels of similar appearance encouraged to take similar depth prediction, but two superpixels in consecutive frames corresponding to the same object are also required to have compatible depth values. To establish temporal correspondence between superpixels in consecutive frames, temporally consistent superpixel (TSP)[14] is employed for video segmentation. Compared with traditional video-based depth estimation approaches such as structure from motion[6], our approach does not rely on parallax introduced by camera motion, and is thus able to process videos containing dynamic scenes or shot by a rotating camera; compared with single-image based CNN depth estimation approaches, our method produces more accurate and consistent depth map sequences required by applications such as 2D to 3D video conversion. The performance of the proposed approach is evaluated with the standard NYU v2 depth dataset as well as the LYB 3D-TV dataset which is collected and annotated by ourselves for 3D-movie generation, demonstrating satisfactory performance for both depth estimation accuracy and temporal consistency.

The rest of the paper is organized as follows. Section 2 reviews the related work in depth estimation for single images and videos, Section 3 introduces our spatial-temporal consistent depth map estimation approach, Section 4 presents our experimental results, and a conclusion is drawn in Section 5.

2 Related Work

There is a long history of work on depth estimation from videos. Among them, approaches based on structure from motion (SfM)[15-16] are the most popular. Given a static scene with sufficient camera motion, SfM is able to obtain camera parameters and sparse 3D point structures from a monocular video sequence. Based on the estimated camera parameters and the

sparse 3D structure, dense depth maps can be obtained by applying multi-view stereo techniques[6,17]. However, these methods generally assume that the video is acquired by a camera whose motion introduces sufficient parallax. When dealing with independently moving objects or static/rotational cameras, these methods typically fail.

To address this limitation, many research studies focus on the estimation of depth from a single static image which eliminates the assumptions on camera motion and scene statics. In this domain, while initial approaches exploit specific cues such as scene geometry[7] and shading[8], the focus has been recently shifted towards employing machine learning methods due to the heavily restrictive assumptions of cue-based methods. In particular, the pioneering work of [18] and [19] modeled depth estimation with a Markov random field. However, they used low-level hand-crafted image features which are insufficient to convey depth information.

In recent years we have witnessed the prosperity of deep convolutional neural networks (CNNs) which have been setting new records for a wide variety of computer vision applications such as image classification[20], object detection[21] and camera pose estimation[22]. A pioneer work of using deep CNN model for depth estimation involves the work of Eigen *et al.*[9], who proposed to train a multi-scale CNN model to directly regress the depth maps from images by optimizing the pixel-wise least square loss. Compared with traditional machine learning approaches based on hand-crafted features, their method drastically improved the depth estimation accuracy when evaluated on several standard image-depth datasets. CNN-based depth estimation approaches have several obvious advantages. First, CNN features are learnt from large-scale data simultaneously with the depth regression task and hence have more expressive power than hand-crafted features, leading to better estimation accuracy. Second, once the model training is finished, predicting the depth of a test image is highly efficient since it does not require iterative optimization. As a result, CNN-based approaches have attracted considerable research interest in depth estimation and many research efforts have been made ever since. In the latter work of Eigen and Fergus[10], they made a deeper CNN network comprised of more spatial scales to achieve higher prediction accuracy, and show that the same network can be applied to other pixel-level prediction tasks such as surface normal prediction and semantic labelling. Li *et al.*[11] proposed to learn the mapping from multi-scale image patches to depth values at the superpixel level, and smooth the CNN output with a CRF model to constrain the spatial coherence of the depth estimations. Their experimental results have shown that by incorporating CRF constraints, their model produces depth estimation accuracy comparable to the work of [9] by using a much smaller training set. However, the CNN and CRF models are used as two disjoint parts in this approach and are trained separately. Very recently, Liu *et al.*[12-13] showed that CNN and CRF can be combined into a uniform network and trained in an end-to-end fashion. The CNN features are hence optimally compatible with the CRF depth estimation model, which brings further improvement in depth estimation accuracy.

To deal with the temporal consistency problem of video-based depth estimation of dynamic scenes acquired by non-translational cameras, Karsch *et al.*[23-24] proposed a sampling-based depth estimation approach for single images and extended it to videos by incorporating temporal consistency constraints. However, their non-parametric model is less competitive in terms of depth estimation accuracy as well as computation efficiency compared with recent CNN-based approaches.

There are also several existing literatures working on the combination of CNN and CRF models, especially for semantic segmentation tasks. In [25], a multi-scale CNN network for scene labelling was proposed, and CRF was used as a post-processing step for local refinement. More recently, Zheng *et al.*[26] formulated mean-field approximate inference for the conditional random fields (CRF) with Gaussian pairwise potentials as recurrent neural networks (RNNs), thereby it can be integrated into the CNN network as a layer and the whole network can be trained in an end-to-end fashion. In these methods, the prediction variables are discrete semantic labels, and thus approximated inference is required. In contrast, our depth estimation task performs continuous variable prediction. The log-likelihood optimization can be directly solved without using approximations since they can be analytically calculated. Moreover, during depth estimation, the estimated depth has a closed-form solution to the MAP inference problem.

3 Our Approach

In the proposed approach, we go beyond single image depth estimation and constrain the consistency of estimated depth maps in both spatial and temporal do-

main by exploring the power of both deep CNN and CRF.

3.1 General Model Structure

The model structure of our video-based depth estimation approach is sketched in Fig.2. Assume that a continuous video clip comprised of m frames is denoted by $\boldsymbol{I} = (\boldsymbol{I}_1, \ldots, \boldsymbol{I}_m)$. These frames are first segmented into temporally consistent superpixels (TSPs)[14], which establish correspondence relationships of superpixels in consecutive frames corresponding to the same object. Each frame is then independently fed into a deep CNN network (SP depth network) parameterized by $\boldsymbol{W}$ to regress a single depth value for each superpixel. The output depth estimation for each single frame $\boldsymbol{I}_t$ is thus represented by an $n_t \times 1$ vector $\boldsymbol{z}_t$, where n_t is the number of superpixels in the frame, and the $\boldsymbol{z}_t$s of all frames are concatenated into a single vector $\boldsymbol{z}$. This raw superpixel depth estimation $\boldsymbol{z}$ as well as the similarity relationships of both spatially neighbouring superpixels ($S^{(\mathrm{s})}$) and temporally corresponding superpixels ($S^{(\mathrm{t})}$) is then fed into a novel CRF layer, and the similarity relationships explicitly model the spatial and temporal depth smoothness and output a refined superpixel depth estimation vector $\hat{\boldsymbol{d}}$. This CRF layer is parameterized by $\boldsymbol{\alpha} = (\boldsymbol{\alpha}^{(\mathrm{s})}, \boldsymbol{\alpha}^{(\mathrm{t})})$ where $\boldsymbol{\alpha}^{(\mathrm{s})}$ and $\boldsymbol{\alpha}^{(\mathrm{t})}$ control the contributions of spatial and temporal similarity constraints of superpixel pairs respectively. Finally, those superpixel depths are projected back to the image domain and constitute the final estimated depth maps. Both the CNN parameters $\boldsymbol{W}$ and the CRF parameter $\boldsymbol{\alpha}$ are optimized simultaneously by back propagation and gradient descent. Our approach extends the deep convolutional neural fields (DCNF) model[12] for single image depth estimation to video domain, and we thus refer to our new model as spatial-temporal DCNF. In Subsection 3.2∼Subsection 3.4, we present different components of our model in detail.

3.2 Temporally Consistent Superpixels (TSP) Segmentation

The goal of depth estimation tasks is to infer the depth of each pixel in a single image. However, this will result in huge output space (number of pixels in the input image) and significantly increase the number of parameters in the CNN network. As a result, several methods such as [9] and [10] only output severely downsampled depth maps, causing blurred depth boundaries. Following the idea in [13], we make the assumption that each individual frame of the video is composed of small homogeneous regions (superpixels), and predict a single depth value for each individual superpixel. By doing so, the number of output variables is significantly reduced while the original resolution of depth maps is maintained, and the resulting depth boundaries are well aligned with the input images as well.

In [13], SLIC algorithm[27] is performed on single

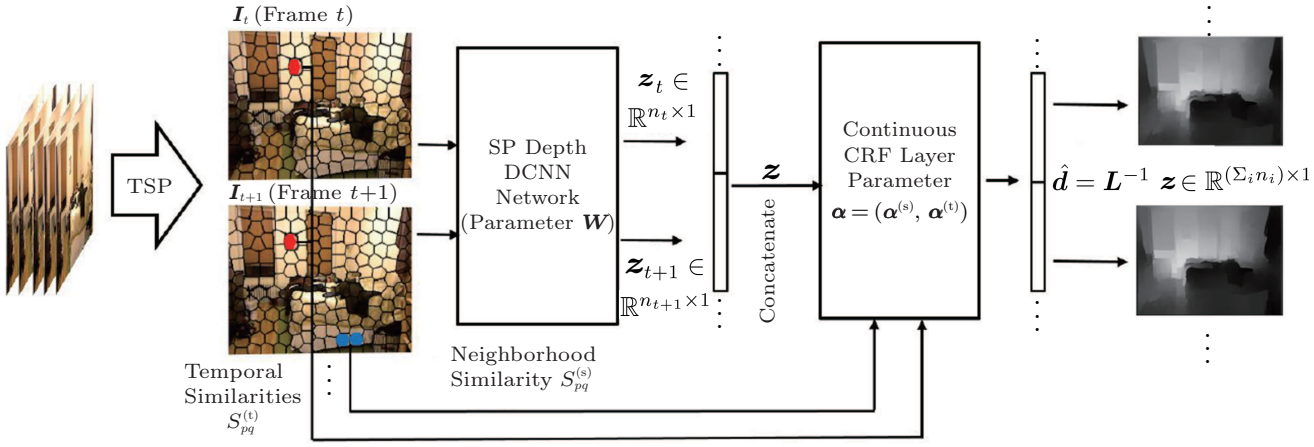


Fig. 2. Illustration of our spatial-temporal DCNF model for depth estimation. The input video is first segmented into temporally consistent superpixels using the approach described in [27]. Each individual frame is passed into a superpixel depth network described in [13] and each superpixel is regressed to a single depth value as the unary part. In the pairwise part, we calculate a similarity measure for each pair of neighboring superpixels (p, q) in a single frame (denoted as $S^{(\mathrm{s})}_{pq}$) and for each pair of corresponding superpixels in two consecutive frames (denoted as $S^{(\mathrm{t})}_{pq}$). The outputs of the SP depth network and the pairwise similarities are then fed to the CRF structured loss layer, which minimizes the negative log-likelihood. Predicting the depth $\hat{\boldsymbol{d}}$ of a new image sequence $\boldsymbol{I}$ is to maximize the conditional probability $\Pr(\boldsymbol{d}|\boldsymbol{I})$, which has a closed-form solution.

images for superpixel segmentation. When applied on consecutive frames in videos, however, even minor variation of input images may cause completely different segmentation results, which contributes to the temporal discontinuity of depth map estimations. By contrast, we apply temporally consistent superpixels (TSP)[14] algorithm on the whole image sequence for segmentation. TSP is a generative probabilistic model for temporally consistent superpixel segmentation of image sequences. Object parts in different frames are tracked by the same temporal superpixel, and those temporal superpixels are kept similar in appearance in each frame. The benefits brought about by TSP are two-fold: first, the temporal continuity of consecutive image frames leads to similar TSP segmentation layouts, which introduce less perturbation to the depth estimation, and more importantly, TSP also generates the temporal correspondence of superpixels in different frames, which naturally constitutes the temporal pairwise constraints required by the CRF layer of our model, which will be introduced later in Subsection 3.4.

3.3 SP Depth Network

After the TSP segmentation, each video clip is fed into a DCNN network which regresses a single depth value for each superpixel. We refer to this network as SP depth network, and illustrate its structure in Fig.3. The network is comprised of seven convolution (conv.) blocks, with the first five blocks identical to the first 31 layers (from the first conv. layer to the 5th pooling layer) of the popular VGG-16 network trained on the ImageNet[28]. In convolution block 6, two more conv. layers are added, followed by a superpixel pooling layer introduced in [13], which performs average pooling to its input feature map within each superpixel

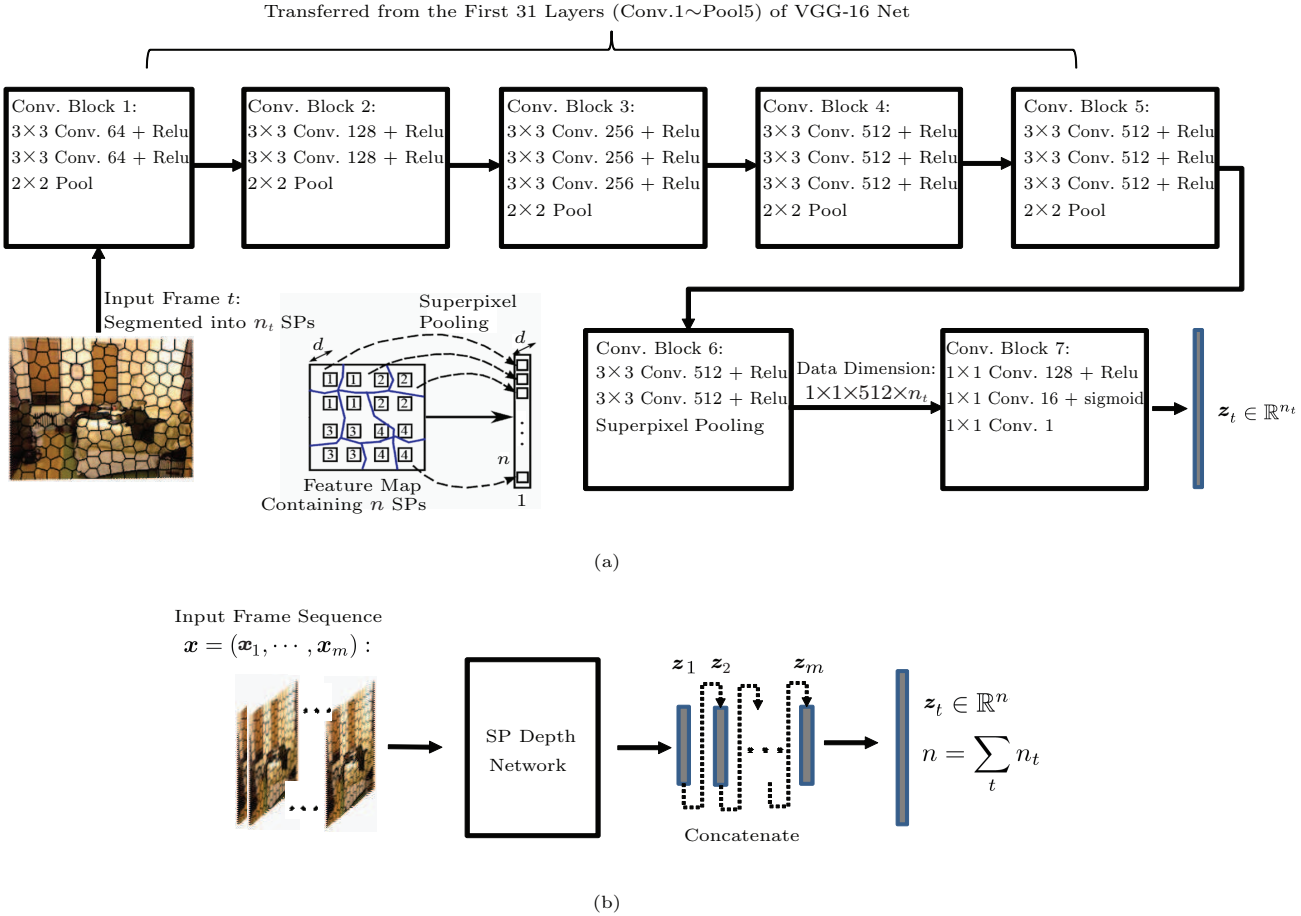


Fig. 3. Illustration for the SP depth network structure. (a) SP depth network structure. The network takes a single image as input and a superpixel depth vector as output, with each of its elements representing the raw estimated depth for a single superpixel of the image. The network is comprised of all conventional CNN layers but a superpixel pooling layer, which does average pooling to the region of a feature map corresponding to a single superpixel into one unique value. (b) Forward pass of an image sequence. When taken an image sequence as input, the network concatenates the SP depth vectors of each frame into a longer output vector.

region and outputs a feature vector for each superpixel. The final convolution block 7 contains three fully connected layers to regress a single depth value for each superpixel. Taking one single frame $\boldsymbol{I}_t$ as input, the SP depth network outputs an n_t-dimensional depth estimation vector $\boldsymbol{z}_t$ where n_t is the number of superpixels in the frame. In our case, the network takes an image sequence of m frames as a mini-batch and outputs a vector $\boldsymbol{z}$ which is the concatenation of all $\boldsymbol{z}_t$s of all frames.

3.4 Spatial-Temporal CRF

3.4.1 Energy Function

The SP depth network gives raw estimates of each superpixel in video frames, but neither spatial nor temporal consistency of the estimated depth maps is considered. A spatial-temporal CRF layer is thus introduced to incorporate these constraints into the DCNN model. We formulate the depth estimation task as a structured-regression task and denote by $\boldsymbol{I}$ one input video and $\boldsymbol{d}$ a vector of continuous depth values corresponding to all n superpixels in $\boldsymbol{I}$. The conditional likelihood for one video is formulated as:

$$P(\boldsymbol{d}|\boldsymbol{I}) = \frac{1}{Z(\boldsymbol{I})}\exp(-E(\boldsymbol{d},\boldsymbol{I})), \tag{1}$$

where Z is the partition function, defined as $Z(\boldsymbol{I}) = \int_{\boldsymbol{d}} \exp\{-E(\boldsymbol{d},\boldsymbol{I})\}\mathrm{d}\boldsymbol{d}$. The energy function $E(\boldsymbol{d},\boldsymbol{I})$ is formulated as:

$$E(\boldsymbol{d},\boldsymbol{I}) = \sum_{p\in\mathcal{N}} U(d_p,\boldsymbol{I}) + \sum_{(p,q)\in\mathcal{S}} V^{(\mathrm{s})}(d_p,d_q,\boldsymbol{I}) + \sum_{(p,q)\in\mathcal{T}} V^{(\mathrm{t})}(d_p,d_q,\boldsymbol{I}), \tag{2}$$

where d_p is the depth of the p-th superpixel in the video, $\mathcal{S}$ is the union of spatially neighbouring superpixel pairs in single frames, and $\mathcal{T}$ is the union of superpixel pairs located in two consecutive frames tracked by the TSP algorithm. This energy function is comprised of a unary term U, a spatial pairwise term $V^{(\mathrm{s})}$, and a temporal pairwise term $V^{(\mathrm{t})}$, which are discussed respectively in the following.

Unary Potential. The unary term U aims to minimize the least-square loss between raw estimated depth values estimated by the SP depth network and the true depth values, which is defined as:

$$U(d_p,\boldsymbol{I};\boldsymbol{W}) = (d_p - z_p(\boldsymbol{W}))^2, \forall p = 1,\ldots,n, \tag{3}$$

where z_p is the raw depth estimation of the p-th temporal superpixel produced by the SP depth network introduced in Subsection 3.3, and is determined by the CNN parameters $\boldsymbol{W}$.

Spatial Pairwise Potential. The spatial pairwise term $V^{(\mathrm{s})}$ encourages spatially neighbouring superpixels with similar appearance to take similar depths, and has the following form:

$$V^{(\mathrm{s})}(d_p,d_q,\boldsymbol{I};\alpha) = \frac{1}{2}\alpha^{(\mathrm{s})}S^{(\mathrm{s})}_{pq}(d_p-d_q)^2, \quad \forall p,q\in\{1,2,\ldots,n\}, \tag{4}$$

where $S^{(\mathrm{s})}_{pq}$ is a visual similarity score between two spatially neighbouring superpixels computed from their color histogram and LBP features, and $S^{(\mathrm{s})}_{pq} = 0$ if superpixels p and q are not neighbours. $\alpha^{(\mathrm{s})}$ is a weighting parameter which needs to be learned from training data.

Temporal Pairwise Potential. To produce temporally consistent depth maps, we also constrain superpixels corresponding to the same object to take similar depth estimation. Similar to the spatial pairwise potential in (4), the temporal pairwise potential $V^{(\mathrm{t})}$ is defined as:

$$V^{(\mathrm{t})}(d_p,d_q,\boldsymbol{I};\alpha^{(\mathrm{t})}) = \frac{1}{2}\alpha^{(\mathrm{t})}S^{(\mathrm{t})}_{pq}(d_p-d_q)^2, \quad \forall p,q\in\{1,2,\ldots,n\}. \tag{5}$$

The TSP algorithm generates temporal correspondence between superpixels in consecutive frames. In (5), $S^{(\mathrm{t})}_{pq} = 1$ if p and q are a pair of superpixels tracked by TSP in two consecutive frames and $S^{(\mathrm{t})}_{pq} = 0$ otherwise. $\alpha^{(\mathrm{t})}$ is a weighting parameter which needs to be learned in the training process.

By substituting U, $V^{(\mathrm{s})}$ and $V^{(\mathrm{t})}$ in (2) by their expressions in (3)$\sim$(5), the energy function can be rewritten as:

$$\begin{aligned} &E(\boldsymbol{d},\boldsymbol{I}) \\ &= \sum_{p\in\mathcal{N}}(d_p - z_p)^2 + \\ &\quad \frac{1}{2}\sum_{(p,q)\in\mathcal{S}}(\alpha^{(\mathrm{s})}S^{(\mathrm{s})}_{pq} + \alpha^{(\mathrm{t})}S^{(\mathrm{t})}_{pq})(d_p-d_q)^2 \\ &= \boldsymbol{d}^{\mathrm{T}}\boldsymbol{L}\boldsymbol{d} - 2\boldsymbol{z}^{\mathrm{T}}\boldsymbol{d} + \boldsymbol{z}^{\mathrm{T}}\boldsymbol{z}, \end{aligned} \tag{6}$$

where $\boldsymbol{z}$ is a vector concatenation of all the $\boldsymbol{z_p}$s, and

$$\boldsymbol{L} = \mathcal{I} + \boldsymbol{D} - \boldsymbol{M},$$
$$\boldsymbol{M} = \alpha^{(\mathrm{s})}\boldsymbol{S}^{(\mathrm{s})} + \alpha^{(\mathrm{t})}\boldsymbol{S}^{(\mathrm{t})},$$

in which $\boldsymbol{S}^{(\mathrm{s})}$ and $\boldsymbol{S}^{(\mathrm{t})}$ are $n\times n$ matrices comprised of $S^{(\mathrm{s})}_{pq}$ and $S^{(\mathrm{t})}_{pq}$ respectively; $\boldsymbol{D}$ is a diagonal matrix with $D_{pp} = \sum_q R_{pq}$. $\mathcal{I}$ is an identity matrix.

Because $E(\boldsymbol{I}, \boldsymbol{d})$ is a quadratic function in terms of continuous depth $\boldsymbol{d}$, the partition function $Z(\boldsymbol{I})$ in (2) can be analytically calculated as:

$$\begin{aligned} Z(\boldsymbol{I}) &= \int_{\boldsymbol{d}} \exp\{-E(\boldsymbol{d}, \boldsymbol{I})\} \mathrm{d}\boldsymbol{d} \\ &= \frac{\pi^{\frac{n}{2}}}{|\boldsymbol{L}|^{\frac{1}{2}}} \exp(\boldsymbol{z}^{\mathrm{T}} \boldsymbol{L}^{-1} \boldsymbol{z} - \boldsymbol{z}^{\mathrm{T}} \boldsymbol{z}), \end{aligned} \tag{7}$$

where $|\boldsymbol{L}|$ is the determinant of matrix $\boldsymbol{L}$.

Substituting (6) and (7) into (1), the conditional likelihood can be written as:

$$\begin{aligned} P(\boldsymbol{d}|\boldsymbol{I}) &= \frac{1}{Z(\boldsymbol{I})} \exp(-E(\boldsymbol{d}, \boldsymbol{I})) \\ &= \frac{|\boldsymbol{L}|^{\frac{1}{2}}}{\pi^{\frac{n}{2}}} \exp(-\boldsymbol{d}^{\mathrm{T}} \boldsymbol{L} \boldsymbol{d} + 2\boldsymbol{z}^{\mathrm{T}} \boldsymbol{d} - \boldsymbol{z}^{\mathrm{T}} \boldsymbol{L}^{-1} \boldsymbol{z}). \end{aligned} \tag{8}$$

3.4.2 Objective and Learning

The whole spatial-temporal DCNF network has two sets of trainable parameters: CNN parameter $\boldsymbol{W}$ for the SP depth network and CRF weighting parameters $\alpha^{(\mathrm{s})}$ and $\alpha^{(\mathrm{t})}$. We use the negative log-likelihood as the loss function for training our model:

$$\underset{\boldsymbol{W}, \alpha^{(\mathrm{s})}, \alpha^{(\mathrm{t})}}{\operatorname{argmin}} - \sum_{i=1}^{N} \log P(\boldsymbol{d}^{(i)} | \boldsymbol{I}^{(i)}), \tag{9}$$

where N is the number of training video clips and

$$\begin{aligned} -\log P(\boldsymbol{d}|\boldsymbol{I}) = &-\boldsymbol{d}^{\mathrm{T}} \boldsymbol{L} \boldsymbol{d} + 2\boldsymbol{z}^{\mathrm{T}} \boldsymbol{d} - \boldsymbol{z}^{\mathrm{T}} \boldsymbol{L}^{-1} \boldsymbol{z} - \\ &\frac{1}{2} \log(|\boldsymbol{L}|) + \frac{n}{2} \log(\pi), \end{aligned}$$

where n is the total number of superpixels in all training videos.

We use stochastic gradient descent (SGD) and back propagation to solve the optimization problem in (9) for learning all parameters of the whole network. The gradients of $-\log \Pr(\boldsymbol{d}|\boldsymbol{I})$ with respect to CNN parameters $\boldsymbol{W}$ can be analytically calculated as:

$$\begin{aligned} \frac{\partial\{-\log \Pr(\boldsymbol{d}|\boldsymbol{I})\}}{\partial \boldsymbol{W}} &= \frac{\partial\{-\log \Pr(\boldsymbol{d}|\boldsymbol{I})\}}{\partial \boldsymbol{z}} \frac{\partial \boldsymbol{z}}{\partial \boldsymbol{W}} \\ &= 2(\boldsymbol{L}^{-1}\boldsymbol{z} - \boldsymbol{d})^{\mathrm{T}} \frac{\partial \boldsymbol{z}}{\partial \boldsymbol{W}}. \end{aligned}$$

For the CRF part, the partial derivative of $-\log \Pr(\boldsymbol{d}|\boldsymbol{I})$ with respect to $\alpha^{(\mathrm{s})}$ and $\alpha^{(\mathrm{t})}$ can be calculated by:

$$\begin{aligned} \frac{\partial\{-\log \Pr(\boldsymbol{d}|\boldsymbol{I})\}}{\partial \alpha^{(s/t)}} = &\boldsymbol{d}^{\mathrm{T}} \boldsymbol{J} \boldsymbol{d} - \boldsymbol{z}^{\mathrm{T}} \boldsymbol{L}^{-1} \boldsymbol{J} \boldsymbol{L}^{-1} \boldsymbol{z} - \\ &\frac{1}{2} Tr(\boldsymbol{L}^{-1} \boldsymbol{J}), \end{aligned}$$

where $Tr()$ is the trace of a matrix, and $\boldsymbol{J}$ is a matrix representing the partial derivative of $\boldsymbol{L}$ with respect to $\alpha^{(\mathrm{s})}$ or $\alpha^{(\mathrm{t})}$ with its elements calculated by:

$$J_{pq} = \frac{L_{pq}}{\alpha^{(s/t)}} = -S_{pq}^{(s/t)} + \delta(p = q) \sum_{q} S_{pq}^{(s/t)},$$

where $\delta(\cdot)$ is the indicator function, which equals 1 if $p = q$ and 0 otherwise.

3.4.3 Prediction

To predict the depth of a new video $\boldsymbol{I}$, the maximum a posterior inference can be performed, which is written as:

$$\hat{\boldsymbol{d}} = \underset{\boldsymbol{d}}{\operatorname{argmax}} \Pr(\boldsymbol{d}|\boldsymbol{I}) = \underset{y}{\operatorname{argmin}} E(\boldsymbol{d}, \boldsymbol{I}). \tag{10}$$

With the energy formulation in (2), we can obtain the closed-form solution for (10):

$$\hat{\boldsymbol{d}} = \boldsymbol{L}^{-1} \boldsymbol{z}. \tag{11}$$

Here $\boldsymbol{L}^{-1}$ can be obtained by solving a linear equation system, and does not need to compute the matrix inverse explicitly.

3.5 Implementation Details

The whole network is implemented in MATLAB based on the efficient CNN toolbox: VLFeat MatConvNet[29]. The network training is performed on a standard desktop with an NVIDIA GTX Titan Balck with 12 GB memory. The parameters of the first five convolution blocks in the SP depth network (shown in Fig.3) are initialized using the Oxford VGG16 network[28]. The learning rate is initialized at 10^{-5} for layers transferred from VGG net and 10^{-4} for the rest of the layers. The momentum is set to 0.9 and the weight decay parameters are set to 0.000 5.

In terms of depth estimation speed, the major computational overhead lies in the TSP segmentation process, which involves optical flow computation between every two consecutive frames and temporally consistent segmentation optimization. Using the MATLAB implementation of TSP provided by the authors of [14], it takes on average eight seconds to segment a frame of 640×480 pixel resolution. After that, a forward pass of frames through the proposed CNN-CRF depth estimation network is relatively fast, averaging 1.5 seconds per frame.

4 Experiments

In this section, the proposed spatial-temporal DCNF approach is evaluated on the public NYU v2 depth dataset[30] as well as on LYB 3D-TV dataset for automatic 3D movie generation which is collected and annotated by ourselves. We experimentally show two major advantages of the proposed approach: 1) our method improves the depth estimation accuracy by incorporating spatial and temporal constraints into the CNN depth estimation network; 2) the temporal consistency of estimated depth maps for continuous image sequences is significantly improved compared with single-image CNN-based methods. The quantitative evaluation is thus done in two parts, namely depth estimation accuracy and temporal consistency.

In terms of depth estimation accuracy, we adopt several measures commonly used in prior work, which include:

• root mean square error (rms):

$$\sqrt{\frac{1}{T}\sum_{p}(\hat{d}_p - d_p)^2},$$

• average log10 error (log10):

$$\frac{1}{T}\sum_{p}|\log_{10}\hat{d}_p - \log_{10} d_p|,$$

• average relative error (rel):

$$\frac{1}{T}\sum_{p}|\hat{d}_p - d_p|/d_p, \text{ and}$$

• accuracy of threshold t:

$$\begin{array}{l}\text{percentage of } \hat{d}_p \\ \text{s.t. } \max(\hat{d}_p/d_p, d_p/\hat{d}_p) < t,\end{array}$$

where $\hat{d}_p$ and d_p are the estimated and the ground truth depth of pixel p respectively, and T is the total number of pixels of all evaluated images.

In existing literatures, we are not able to identify an appropriate measure to quantitatively evaluate the temporal consistency of depth estimation methods. Intuitively, to ensure temporal continuity, the difference between the estimated depths of a pair of pixels in two consecutive frames corresponding to the same 3D points should be equal to the difference of their ground truth depths. In the TSP algorithm, object parts in two consecutive frames are tracked by the same temporal superpixel. The temporal consistency of the depth estimation of a superpixel p (denoted as $\hat{d}_p$) can be thus measured by its relative temporal error (RTE) to its corresponding superpixel in the previous frame (denoted as $\hat{d}_{p(-1)}$):

$$RTE(p) = \frac{|\hat{d}_p - \hat{d}_{p(-1)} - (d_p - d_{p(-1)})|}{d_p}. \quad (12)$$

For the evaluation of a whole video, we use two measures:

• average RTE: $\frac{1}{T}\sum_{p\in\mathcal{T}} RTE(p)$, and

• RTE percentage of threshold t: percentage of superpixel p s.t. $RTE(p) > t$,

as evaluation metrics for temporal consistency. Note that superpixels which have been lost tracking by the TSP algorithm are not included in the calculation of the measures.

4.1 NYU v2 Dataset

Our proposed method is first evaluated on the publicly available NYU Depth v2 dataset[30], which is composed of indoor scenes taken as video sequences using a Microsoft Kinect camera with a resolution of 640×320. Following the standard split of the dataset, 795 scenes are used for training and 654 for test. For each scene, we use 30 temporally consecutive image-depth pairs as a video clip. The TSP[14] algorithm is applied to each video clip to segment them into temporally consistent superpixel. In the TSP algorithm, there are mainly two hyper-parameters to be manually set: hyper-parameter M which controls the designed superpixel number per frame, and α which controls the superpixel shape regularity. According to our experiments, varying M from 500 to 1 000 does not have notable influence on depth estimation accuracy but has an impact on the training/testing speed and the coarseness of the estimated depth maps. We select $M = 750$ for a good compromise. For hyper-parameter α, several test runs are performed on some sample image sequences and α could be chosen to be a value which could generate regular shaped superpixels. Since the ground truth depth maps contain empty values, those superpixels which contain no ground truth depth value are masked out during back propagation and are not used for building pairwise similarity graphs. To remove many invalid regions caused by windows, open doorways and specular surfaces, we also mask out depths equal to the minimum or maximum recorded for each image.

We compare the proposed spatial-temporal DCNF with two state-of-the-art CNN-based approaches, namely multi-scale CNN[9] and 2D-DCNF[13], as well

as two other recently proposed non-CNN-based approaches, depth transfer[24] and discrete-continuous CRF[31]. The depth estimation accuracy and temporal consistency performances of the compared algorithms are summarized in Table 1 and Table 2. Note that the 2D-DCNF model reported in [13] is trained on 795 single images (one image per scene), which is smaller than our training set (30 consecutive images per scene). For fair comparison, we implemented their method based on the test code released by the authors of [13] and trained a new 2D-DCNF model with the same training set of our approach. The performance of the new 2D-DCNF model is also reported in Table 1. The multi-scale CNN approach[9] used a much larger training set (the entire raw dataset) than ours, thereby we directly cite the results from the paper[9].

In terms of depth estimation accuracy performances denoted in Table 1, the CNN-based approaches ([9], [13] and our approach) significantly out-perform non-CNN-based approaches with relative performance gains over 30%, which demonstrates the power of deep CNN models. Compared with other single-image-based CNN approaches of [9] and [13], our spatial-temporal DCNF approach achieved the best performance in all evaluation metrics. In particular, the comparison of our approach and its single-image-based version 2D-DCNF is especially meaningful, showing the benefits by the incorporation of temporal consistency constraints.

Table 2 compares the temporal consistency performance of our approach with two other single-image-based CNN approaches. For multi-scale CNN[9], the results are obtained by applying the prediction model to our test set and upsampling the resulting depth maps to the original input image size of 640×480, while for 2D-DCNF[13], both the prediction model provided by the authors of [13] (trained on 795 single images) and the model trained by ourselves on the augmented training set are evaluated. It is observed that the 2D-DCNF approach achieves the worse performance in temporal continuity. When trained with more training samples, the temporal consistency errors are not reduced significantly. The multi-scale CNN approach has a lower temporal continuity error, partly due to the low resolution of estimated depth maps (only 74×55 pixels in size) and the error is smoothed out in the up-sampling process. Finally, the proposed spatial-temporal DCNF approach achieved significantly lower error in all temporal consistency metrics, and for the RTE percentage of threshold t measures, our error is one order of magnitude smaller than its single-image counterpart 2D-DCNF.

In Fig.4, we provide a qualitative comparison of our approach and [13] on a living room scene in the test set of NYU v2 dataset. The estimated depth maps for five consecutive frames by [11] and our approach are shown in Fig.4(c) and Fig.4(d) respectively. For each frame, the relative temporal error of each superpixel calculated with (12) is projected back to its corresponding region and visualized as heat-maps, as shown in Figs.4(e) and 4(f). Many red and yellow regions are observed in the RTE heat-maps of 2D-DCNF, representing up to 20% of relative temporal difference in depth estimation with respect to the ground truth depths. Our approach, on

Table 1. Precision Performance Comparisons on the NYU v2 Dataset

Method	Error (Lower Is Better)			Accuracy of Threshold t (Higher Is Better)		
	rel	log10	*rms*	$t = 1.25$	$t = 1.25^2$	$t = 1.25^3$
Depth transfer[24]	0.350	0.131	1.200	-	-	-
Discrete-continuous CRF[31]	0.335	0.127	1.060	-	-	-
Multi-scale CNN[9]	0.245	-	0.907	0.611	0.887	0.971
2D-DCNF[13]	0.213	0.087	0.759	0.650	0.906	0.976
2D-DCNF[13] (augmented training set)	0.202	0.088	0.730	0.659	0.911	0.980
Spatial-temporal DCNF (ours)	**0.188**	**0.085**	**0.699**	**0.690**	**0.923**	**0.985**

Table 2. Temporal Consistency Performance Comparisons on the NYU v2 Dataset

Method	Mean RTE (Lower Is Better)	RTE % of Threshold t (Lower Is Better)		
		$t = 0.03$	$t = 0.05$	$t = 0.10$
Multi-scale CNN[9]	0.019	0.088	0.025	0.012
2D-DCNF[13]	0.035	0.164	0.084	0.031
2D-DCNF[13] (augmented training set)	0.032	0.171	0.082	0.029
Spatial-temporal DCNF (ours)	**0.011**	**0.014**	**0.007**	**0.002**

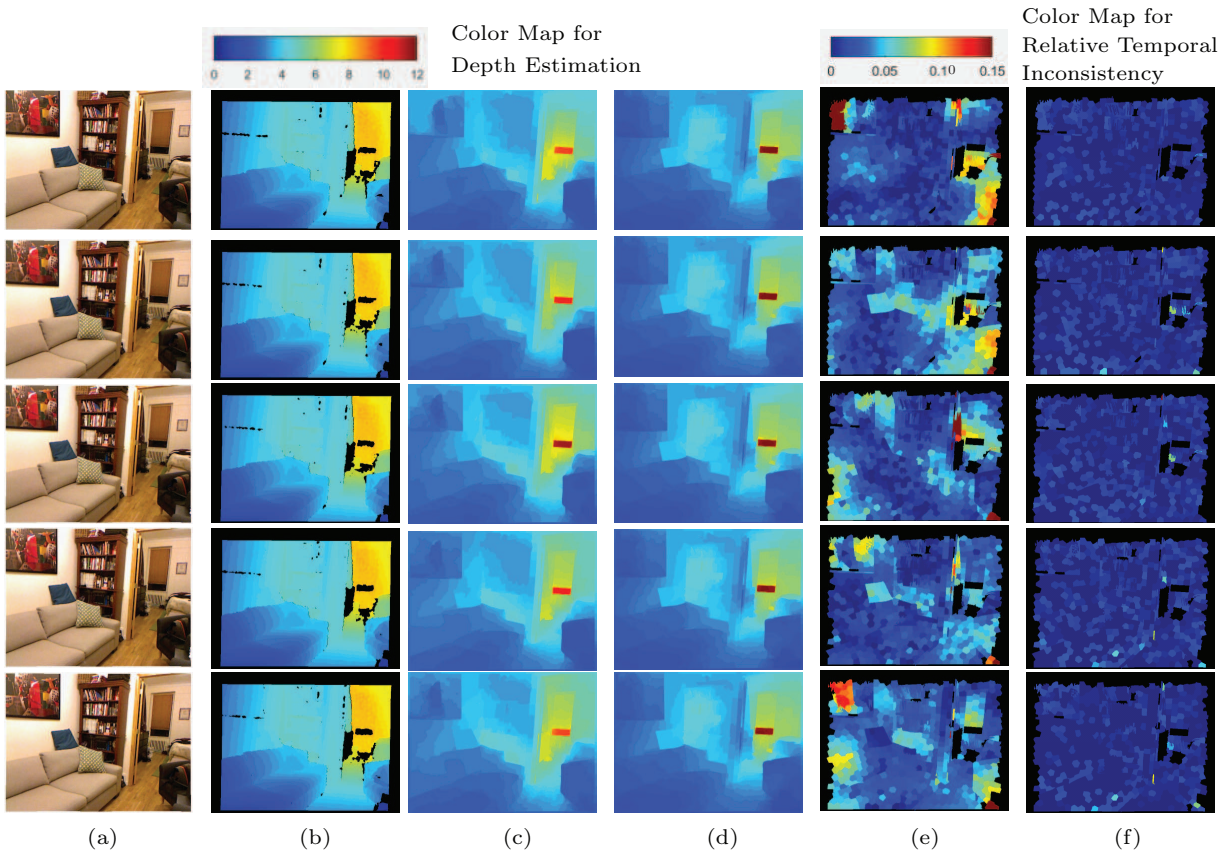


Fig. 4. Depth map estimations for 5 consecutive frames of a living room scene in NYU v2 dataset generated by our spatial-temporal DCNF and 2D-DCNF[13]. To better visualize the temporal continuity qualities, the relative temporal errors for all temporal superpixels are projected to their locations in the corresponding frames and the resulting images are plotted as heat-maps. It is shown that our approach produces much lower RTE than 2D-DCNF. (a) RGB input. (b) Ground truth depth. (c) Depth estimated by [13]. (d) Depth estimated by our approach. (e) RTE of [13]. (f) RTE of our approach.

the other hand, produces much more temporally consistent depth estimations with RTE no more than 5%.

4.2 LYB 3D-TV Dataset

Generating high quality depth maps for image sequences is a crucial component in 2D to 3D video conversion. Given a 2D video and its corresponding depth maps for each frame, a synthetic viewpoint can be generated using depth image based rendering (DIBR) techniques[32]. To obtain these depth maps, several semi-automatic approaches have been proposed[33-34], and allow the user to make sparse depth annotations on a frame, and the dense depth maps are obtained by propagating these user annotations to the entire frame. Even though this scheme reduces the time consumed in comparison to the pure manual depth annotation, a significant amount of human engagement is still required to complete the conversion. To convert the vast collection of available 2D material into 3D in an economic manner, an automatic depth estimation scheme is desired. To validate the performance of the proposed spatial-temporal DCNF approach in the context of 3D video conversion, we thus made a dataset by making depth annotations of a set of selected scenes from a popular Chinese historical drama called Lang Ya Bang (Nirvana in Fire in English). This dataset is referred to as LYB 3D-TV dataset.

The dataset is comprised of 80 different scenes, and each is a short video clip containing 48∼186 consecutive frames. The whole dataset contains 6 402 individual frames in total, with each frame annotated by a human operator with a semi-automatic depth annotation system to generate a gray-scale monochromatic depth map. Note that the resulting depth values range from 0 to 1, which do not correspond to the real depth value in meters but represent relative depths, with a smaller value indicating a closer location to the viewer. Used as the input of a DIBR system, these annotated depth maps are able to generate plausible synthesized

3D videos for commercial distribution. Several sample frames and their corresponding depth maps are shown in Fig.5. All the frames and depth maps are resized to the resolution of 640×360 for computational efficiency. The whole dataset is divided into a training set of 60 scenes containing 5 124 single frames and a test set of 20 scenes including 1 278 frames.

Fig.5. Several example images from our LYB dataset and their corresponding depth maps.

The depth estimation performance of the proposed 3D-DCNF approach is again evaluated in terms of both estimation accuracy and temporal consistency, and compared with the two CNN-based approaches, multi-scale CNN[9] and 2D-DCNF[13]. The results with the best performance in bold are summarized in Table 3 and Table 4 respectively. In terms of depth estimation accuracy, our method achieved better performance in most of the metrics, only with the root mean square (rms) error slightly larger than 2D-DCNF. Note that the log10 error metric is not used since it aims to compensate the influence of large depth values while our ground truth depth values only vary from 0 to 1. In terms of temporal consistency, our method outperforms the compared approaches in all metrics with significant margins. In Fig.6, a qualitative comparison of our approach and [13] on estimating the depth of five consecutive frames in the test set is provided. It is observed that [13] produces obvious temporal discontinuity at several regions in each frame, which are marked in dashed circles. The regions have high relative temporal error (RTE) up to 45% and correspond to the red and yellow regions in the RTE heat-maps. Our approach, on the other hand, produces much more temporally consistent depth estimations, as shown by the RTE heat-maps in Fig.6(f).

5 Conclusions

This paper presented a novel CNN-based depth estimation approach which is able to produce temporally consistent depth map estimations for videos. In contrast to existing CNN-based depth estimation methods which are optimized on single frames, our approach was optimized over consecutive frames by incorporating a novel spatial-temporal CRF layer into the deep CNN architecture, which is able to enforce both spatial and temporal consistency between estimated depth maps of consecutive frames. Experimental results on the standard NYU v2 dataset and our LYB 3D-TV dataset showed that our approach is able to not only significantly enhance the temporal consistency of estimated depth maps over existing single-frame-based approaches, but also improve the depth estimation accuracy in terms of various evaluation metrics.

One limitation our method concerns is the processing speed. Indeed, the proposed approach is a relatively slow method, with the TSP segmentation and the large

Table 3. Precision Comparisons on the LYB 3D-TV Dataset

Method	Error (Lower Is Better)		Accuracy of Threshold t (Higher Is Better)		
	rel	rms	$t = 1.25$	$t = 1.25^2$	$t = 1.25^3$
Multi-scale CNN[9]	0.264	0.161	0.572	0.810	0.941
2D-DCNF[13]	0.234	**0.132**	0.623	0.860	0.960
Spatial-temporal DCNF (ours)	**0.225**	0.135	**0.648**	**0.889**	**0.968**

Table 4. Temporal Consistency Comparisons on the LYB 3D-TV Dataset

Method	Mean RTE (Lower Is Better)	RTE % of Threshold t (Lower Is Better)		
		$t = 0.05$	$t = 0.10$	$t = 0.20$
Multi-scale CNN[9]	0.035	0.162	0.102	0.025
2D-DCNF[13]	0.078	0.428	0.234	0.099
Spatial-temporal DCNF (ours)	**0.024**	**0.107**	**0.031**	**0.007**

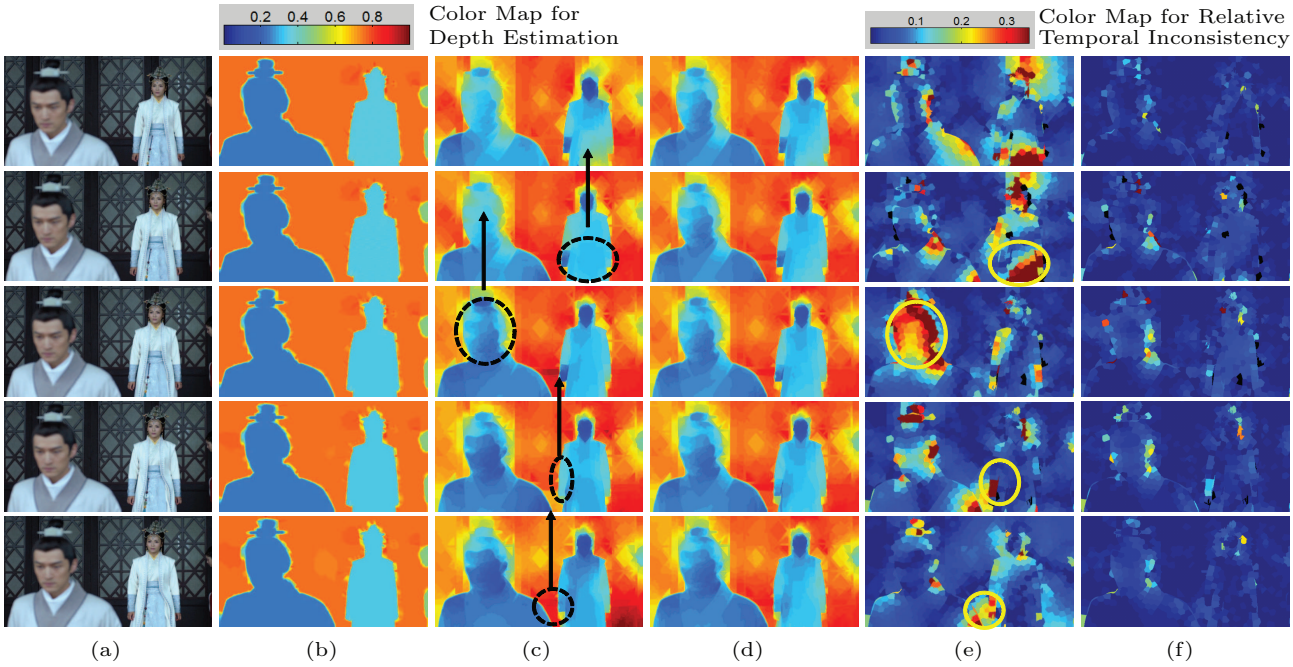


Fig. 6. Depth map estimations for 5 consecutive frames in the test set of LYB 3D-TV dataset by our approach and the 2D-DCNF method. Obvious temporal discontinuities of estimated depth by 2D-DCNF are marked out in dashed ellipses, and these regions have high RTE values up to 45%. The temporal discontinuity of estimated depth maps is significantly reduced in our method, as shown by the RTE heat-maps. (a) RGB input. (b) Ground truth depth. (c) Depth estimated by [13]. (d) Depth estimated by our approach. (e) RTE of [13]. (f) RTE of our approach.

linear systems solution in the inference step being the main computational overhead. One potential solution to this problem is to use temporal models such as Recurrent Neural Network (RNN)[35] or Long-Short Term Memory (LSTM)[36], which can be integrated into the CNN network to model the temporal continuity. That is, the estimated depth map of one frame depends not only on the RGB input of the current frame, but also on the depth estimation outputs of previous frames, and such dependency is learned by RNN or LSTM during training. By doing so, both the time-consuming TSP segmentation and the CRF inference are removed, and such a model can predict depth frame by frame with normal CNN forward pass procedure. Application of such CNN-RNN or CNN-LSTM models to video-based depth estimation comprises our potential future work.

References

[1] Saxena A, Sun M, Ng A. Learning 3-D scene structure from a single still image. In *Proc. the 11th IEEE International Conference on Computer Vision*, October 2007.

[2] Shotton J, Sharp T, Kipman A *et al.* Real-time human pose recognition in parts from single depth images. *Communications of the ACM*, 2013, 56(1): 116-124.

[3] Cheng K L, Ju X, Tong R F, Tang M, Chang J, Zhang J J. A linear approach for depth and colour camera calibration using hybrid parameters. *Journal of Computer Science and Technology*, 2016, 31(3): 479-488.

[4] Fanello S R, Keskin C, Izadi S, Kohli P, Kim D, Sweeney D, Criminisi A, Shotton J, Kang S B, Paek T. Learning to be a depth camera for close-range human capture and interaction. *ACM Transactions on Graphics*, 2014, 33(4): 86:1-86:11.

[5] Zhang L, Vázquez C, Knorr S. 3D-TV content creation: Automatic 2D-to-3D video conversion. *IEEE Transactions on Broadcasting*, 2011, 57(2): 372-383.

[6] Zhang G F, Jia J, Wong T T, Bao H J. Consistent depth maps recovery from a video sequence. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2009, 31(6): 974-988.

[7] Tsai Y M, Chang Y L, Chen L G. Block-based vanishing line and vanishing point detection for 3D scene reconstruction. In *Proc. International Symposium on Intelligent Signal Processing and Communications*, December 2006, pp.586-589.

[8] Zhang R, Tsai P S, Cryer J E, Shah M. Shape-from-shading: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1999, 21(8): 690-706.

[9] Eigen D, Puhrsch C, Fergus R. Depth map prediction from a single image using a multi-scale deep network. In *Proc. Advances in Neural Information Processing Systems*, December 2014, pp.2366-2374.

[10] Eigen D, Fergus R. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proc. IEEE International Conference on Computer Vision*, December 2015, pp.2650-2658.

[11] Li L, Shen C H, Dai Y C, van den Hengel A, He M. Depth and surface normal estimation from monocular images using regression on deep features and hierarchical CRFs. In *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, June 2015, pp.1119-1127.

[12] Liu F, Shen C, Lin G. Deep convolutional neural fields for depth estimation from a single image. In *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, June 2015, pp.5162-5170.

[13] Liu F, Shen C H, Lin G S, Reid I. Learning depth from single monocular images using deep convolutional neural fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016, 38(10): 2024-2039.

[14] Chang J, Wei D, Fisher J. A video representation using temporal superpixels. In *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, June 2013, pp.2051-2058.

[15] Azarbayejani A, Pentland A P. Recursive estimation of motion, structure, and focal length. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1995, 17(6): 562-575.

[16] Pollefeys M, van Gool L V, Vergauwen M, Verbiest F, Cornelis K, Tops J, Koch R. Visual modeling with a hand-held camera. *International Journal of Computer Vision*, 2004, 59(3): 207-232.

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

[18] Saxena A, Chung S, Ng A Y. 3-D depth reconstruction from a single still image. *International Journal of Computer Vision*, 2008, 76(1): 53-69.

[19] Saxena A, Sun M, Ng A. Make3D: Learning 3D scene structure from a single still image. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 2009, 31(5): 824-840.

[20] Krizhevsky A, Sutskever I, Hinton G. Imagenet classification with deep convolutional neural networks. In *Proc. the 26th Advances in Neural Information Processing Systems*, December 2012, pp.1106-1114.

[21] Zhu Z, Liang D, Zhang S, Huang X, Li B L, Hu S M. Traffic-sign detection and classification in the wild. In *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, June 2016, pp.2110-2118.

[22] Nakajima Y, Saito H. Robust camera pose estimation by viewpoint classification using deep learning. *Computational Visual Media*, 2016.

[23] Karsch K, Liu C, Kang S B. Depth extraction from video using non-parametric sampling. In *Proc. European Conference on Computer Vision*, October 2012, pp.775-788.

[24] Karsch K, Liu C, Kang S B. Depth transfer: Depth extraction from video using non-parametric sampling. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 2014, 36(11): 2144-2158.

[25] Farabet C, Couprie C, Najman L, LeCun Y. Learning hierarchical features for scene labeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2013, 35(8): 1915-1929.

[26] Zheng S, Jayasumana S, Romera-Paredes B, Vineet V, Su Z, Du D L, Huang C, Torr P. Conditional random fields as recurrent neural networks. In *Proc. the IEEE International Conference on Computer Vision*, December 2015, pp.1529-1537.

[27] Achanta B, Shaji A, Smith K, Lucchi A, Fua P, Süsstrunk S. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2012, 34(11): 2274-2282.

[28] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. https://arxiv.org/abs/1409.1556, March 2017.

[29] Vedaldi A, Lenc K. MatConvNet: Convolutional neural networks for MATLAB. In *Proc. the 23rd ACM International Conference on Multimedia*, October 2015, pp.689-692.

[30] Silberman N, Hoiem D, Kohli P, Fergus R. Indoor segmentation and support inference from RGBD images. In *Proc. the 12th European Conference on Computer Vision*, October 2012, pp.746-760.

[31] Liu M M, Salzmann M, He X. Discrete-continuous depth estimation from a single image. In *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, June 2014, pp.716-723.

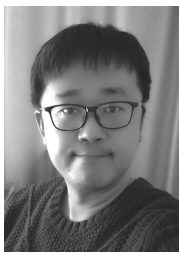
[32] Fehn C, de la Barré R, Pastoor S. Interactive 3-DTV-concepts and key technologies. *Proceedings of the IEEE*, 2006, 94(3): 524-538.

[33] Cao X, Zheng Li, Dai Q H. Semi-automatic 2D-to-3D conversion using disparity propagation. *IEEE Transactions on Broadcasting*, 2011, 57(2): 491-499.

[34] Phan R, Androutsos D. Robust semi-automatic depth map generation in unconstrained images and video sequences for 2D to stereoscopic 3D conversion. *IEEE Transactions on Multimedia*, 2014, 16(1): 122-136.

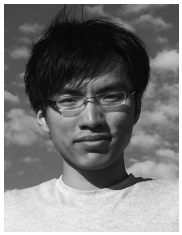
[35] Mikolov T, Kombrink S, Burget L, Cernocky J, Khudanpur S. Extensions of recurrent neural network language model. In *Proc. the IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2011, pp.5528-5531.

[36] Graves A, Mohamed A, Hinton G. Speech recognition with deep recurrent neural networks. In *Proc. International Conference on Acoustics, Speech and Signal Processing*, May 2013, pp.6645-6649.



Xu-Ran Zhao is currently an assistant professor at the School of Computer Science and Information Engineering, Zhejiang Gongshang University, Hangzhou. He received his B.S. degree in electronic and information technologies from Shanghai University, Shanghai, and M.S. degree in electrical and computer engineering from Georgia Institute of Technology, Atlanta, in 2006 and 2010 respectively. He received his Ph.D. degree from Telecom ParisTech, Paris, in 2013. During 2014∼2016, he worked as a postdoctoral researcher on machine learning in School of Computer Science at Aalto University, Helsinki. His current research interests include pattern recognition, computer vision and biometric recognition.

Xun Wang is currently a professor at the School of Computer Science and Information Engineering, Zhejiang Gongshang University, Hangzhou. He received his B.S. degree in mechanics, and Ph.D. degree in computer science, both from Zhejiang University, Hangzhou, in 1990 and 2006, respectively. His current research interests include mobile graphics computing, virtual reality, image/video processing, computer vision, and visual analytics. He has published over 100 papers in high-quality journals and conferences. He a senior member of CCF, and is a member of ACM and IEEE.



Qi-Chao Chen is currently pursuing his Master's degree in engineering at the School of Computer Science and Information Engineering, Zhejiang Gongshang University, Hangzhou. His research interests include video/image processing and pattern recognition.