Cogn Comput (2018) 10:260–271
https://doi.org/10.1007/s12559-017-9526-9

Semantic Scene Mapping with Spatio-temporal Deep Neural Network for Robotic Applications

Ruihao Li[1] · **Dongbing Gu**[1] · **Qiang Liu**[1] · **Zhiqiang Long**[2] · **Huosheng Hu**[1]

Received: 25 September 2017 / Accepted: 31 October 2017 / Published online: 24 November 2017

Abstract Semantic scene mapping is a challenge and significant task for robotic application, such as autonomous navigation and robot-environment interaction. In this paper, we propose a semantic pixel-wise mapping system for potential robotic applications. The system includes a novel spatio-temporal deep neural network for semantic segmentation and a Simultaneous Localisation and Mapping (SLAM) algorithm for 3D point cloud map. Their combination yields a 3D semantic pixel-wise map. The proposed network consists of Convolutional Neural Networks (CNNs) with two streams: spatial stream with images as the input and temporal stream with image differences as the input. Due to the use of both spatial and temporal information, it is called spatio-temporal deep neural network, which shows a better performance in both accuracy and robustness in semantic segmentation. Further, only keyframes are selected for semantic segmentation in order to reduce the computational burden for video streams and improve the real-time performance. Based on the result of semantic segmentation, a 3D semantic map is built up by using the 3D point cloud map from a SLAM algorithm. The proposed spatio-temporal neural network is evaluated on both Cityscapes benchmark (a public dataset) and Essex Indoor benchmark (a dataset we labelled ourselves manually). Compared with the state-of-the-art spatial only neural networks, the proposed network achieves better performances in both pixel-wise accuracy and Intersection over Union (IoU) for scene segmentation. The constructed 3D semantic map with our methods is accurate and meaningful for robotic applications.

Keywords Deep learning · Spatio-temporal neural network · 3D semantic map · Robotics

Introduction

Semantic scene mapping is a challenge and significant task for autonomous navigation, localisation, robot-environment interaction, etc. As it can provide semantic information and understanding of the environments, it is widely investigated in robotics, computer vision, augment reality (AR), and virtual reality (VR). Semantic pixel-wise segmentation is the basis of semantic scene mapping and has gained a great success due to the spectacular development of deep Convolutional Neural Networks (CNNs) in the past few years. The deep CNNs have proved their powerful abilities in many aspects, such as objects recognition [1, 2], scene segmentation [3, 4], and so on.

Electronic supplementary material The online version of this article (https://doi.org/10.1007/s12559-017-9526-9) contains supplementary material, which is available to authorized users.

✉ Ruihao Li
rlig@essex.ac.uk

Dongbing Gu
dgu@essex.ac.uk

Qiang Liu
qliui@essex.ac.uk

Zhiqiang Long
lzq@maglev.cn

Huosheng Hu
hhu@essex.ac.uk

[1] Department of Computer Science and Electronic Engineering, University of Essex, Colchester, CO4 3SQ, UK

[2] College of Mechatronics and Automation, National University of Defense Technology, Changsha, China

Since a fully convolutional network (FCN) [3] was proposed for scene segmentation and achieved the state-of-the-art pixel-wise segmentation performance, researchers have proposed many different kinds of deep convolutional neural network architectures derived from the FCN. These CNNs can learn the spatial information from images and obtain pixel-wise understanding of the environments. But most of them are for static images and time-consuming. They can hardly have a real-time performance. This work aims to producing the semantic pixel-wise segmentation from video streams, which not only contain the spatial information but also the temporal information for potential robotic applications. The temporal information is valuable in the pixel-wise segmentation as pixels in adjacent images have some forms of corresponding geometry constraints. In this paper, we present a novel spatio-temporal neural network for semantic segmentation. In addition to the images as spatial information, the image difference between two consecutive images is used as the temporal information for the network. And the computational burden is reduced by selecting only keyframes for segmentation while the non-keyframe segmentation is predicted by the results from keyframes.

It is known that 3D point cloud maps produced by visual Simultaneous Localisation and Mapping (SLAM) [5, 6] algorithms only represent the occupation information and are less meaningful. However, 3D semantic maps can provide more meaningful information for robotic applications. Semantic pixel-wise segmentation can be combined with a 3D point cloud map to yield a 3D semantic map. In this paper, we also present how to yield a 3D semantic map through this combination. This is achieved by using a SLAM algorithm to construct a 3D point cloud map and then labelling each point in the point cloud by using the semantic segmentation result. Some 3D semantic maps are shown in Fig. 1.

Our main contributions in this paper are summarised as follows:

- We propose a novel spatio-temporal neural network for semantic scene segmentation. Both images and image differences between two consecutive images are taken as the inputs for the network. In this way, both the spatial and temporal information are considered for semantic segmentation.
- We propose to select the keyframes for semantic segmentation while the non-keyframe segmentation is established by the prediction result from keyframes. In this way, the computational burden when handling with video streams can be reduced and the real-time performance can be improved.

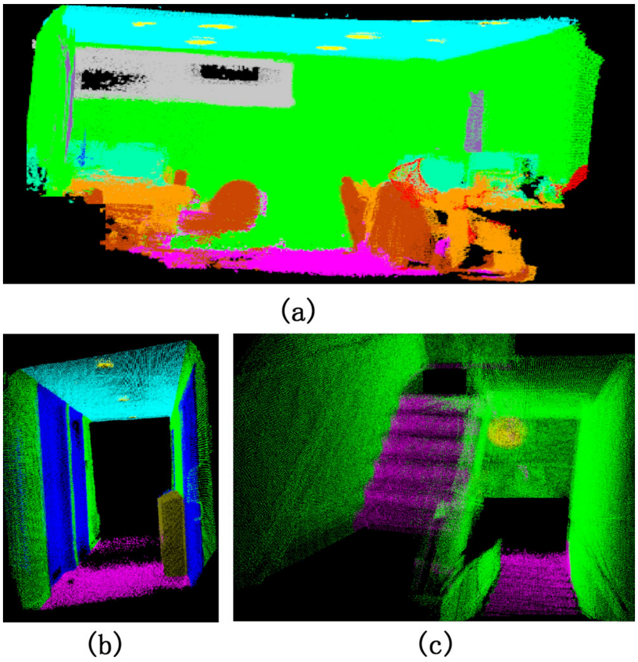

Fig. 1 3D semantic maps. **a** 3D semantic map of a room. **b** 3D semantic map of a corridor. **c** 3D semantic map of a staircase

- We develop a practical semantic scene mapping system by using a visual SLAM algorithm combined with the result from semantic segmentation. Qualitative and quantitative experiments based on the public dataset and our own dataset are presented. The 3D semantic map of our dataset is built up for robotic applications.

In the following section, we will give a review of the related work. In "Approach", we will provide an introduction to the architecture of the proposed CNN for semantic segmentation and present the fusion method for spatial and temporal information. The procedures of keyframe selection, segmentation prediction for non-keyframes, and semantic map construction will be also described in this section. "Experimental Evaluation" will demonstrate the experimental results on different datasets using the proposed system. In "Conclusions", we will give a summary conclusion and the future work we would like to investigate.

Related Work

In this part, we will review the research on semantic scene segmentation using deep CNNs for single stream (spatial only) network. The research on multi-stream networks are followed. Then, we review the research on semantic mapping with SLAM.

Spatial Image Segmentation with CNNs

Semantic segmentation is a traditional field in computer vision [7]. In 2015, Long et al. [3] first proposed to replace fully connected layers with fully convolutional layers in CNNs. In this way, the CNNs have been successfully applied to spatial dense tasks, such as pixel-wise segmentation, depth estimation, and optical flow estimation. Following from VGG net [8], the FCN was trained end-to-end for image segmentation. In order to take the advantage of global context information, Liu et al. [9] introduced the global pooling into FCN. The proposed ParseNet exceeded the FCN in semantic segmentation due to the wider view of the network. Badrinarayanan et al. [10] presented a novel network architecture called SegNet for scene parsing. The SegNet is based on the FCN and has an encoder-decoder architecture. The decoder performs the upsampling for low-resolution features. Afterwards, Kendall et al. [11] added dropout layers to the SegNet and proposed a Beyasian SegNet which could estimate the probability for pixel-level segmentation.

Conditional Random Fields (CRFs) have proved its powerful capability in image segmentation and been adopted as a post-processing method to refine the image segmentation. Zheng et al. [12] proposed to formulate the probabilistic mean field inference with CRFs as Recurrent Neural Networks (RNNs). By embedding CRFs into CNNs, they presented a novel network architecture called CRF-RNN, which combines the strength of both CNNs and CRFs. Arnab et al. [13] designed two high order potentials based on object detection and superpixels later and integrated them into the CRF-RNN. However, CRFs are especially computational intensive and not suitable for real-time applications.

The networks mentioned above all use the VGG [8] as their base network architecture. After He et al. [2] proposed a very deep ResNet, most researchers began to use the ResNet as the basic network architecture. The ResNet demonstrated a astonishing performance in the ImageNet classification challenge [14] and has been widely applied for many tasks. Chen et al. [15–17] proposed to use the very deep ResNet, dilated convolution, and fully connected CRFs to segment images. By using dilated convolution, the field-of-view of filters can be enlarged effectively without increasing the computation. Atrous Spatial Pyramid Pooling (ASPP) and multiple scale technology were also introduced in Deeplab [16], which performed extremely well in PASCAL VOC-2012 [18] semantic image segmentation dataset. Wu et al. [19] explored different variations of the ResNet in order to find the best network configuration, such as the number of layers, the size of field-of-view, and the resolution of feature maps. An online bootstrapping method was also used during training to improve the segmentation performance. The proposed network was evaluated on both PASCAL VOC-2012 benchmark and Cityscapes [20] benchmark. The results showed that the proposed network was very competitive when compared with other methods. Afterwards, Wu et al. [21] further studied the relationship between the depth of residual networks and the performance and proved that some relatively shallow residual networks could outperform much deeper networks, particularly within some limitations. This performance was not only applied to the recognition task but also suitable for the semantic segmentation task. Zhao et al. [4] proposed the Pyramid Scene Parsing Network (PSPNet) which won the ImageNet scene parsing challenge 2016 [22]. Different from the global pooling method proposed in [9], the global spatial context information in images was exploited by different-region-based aggregation with the proposed pyramid pooling model in [4]. Wu et al. [21] and Zhao et al. [4] both used four graphics processing units (GPUs) to train the network, and they could choose large “cropsize” and “batchsize” which are critical for the performance.

Besides, Tu et al. [23] introduced optical flow as the temporal information. By combining this motion-based saliency method with a region-based image saliency method, they demonstrated a spatio-temporal system for object segmentation. Doborjeh [24] made use of spatio-temporal EEG data and used spiking neural networks to realise the classification of signal. Wang et al. [25] proposed DeepVO which used Recurrent Convolutional Neural Network (RCNN) to perform visual odometry (VO). Both spatial and temporal image information were used.

Multi-stream CNNs

With regard to temporal information, Wang et al. [26, 27] proposed a novel temporal segmentation network to exploit the optical flow along with colour images to enhance the performance of action recognition in video streams. The proposed network demonstrated a high performance in the Large Scale Activity Recognition Challenge 2016.

Aiming at the pose regression in challenging indoor environments, Li et al. [28] presented a novel dual-stream CNN architecture to take colour images and disparity images as the inputs at the same time. Eitel et al. [29] and Schwarz et al. [30] rendered disparity images with a colour palette and use a two-stream CNN to obtain a better performance with RGB-D cameras.

Based on the SegNet [10], Hazirbas et al. [31] extracted features from colour images and depth images respectively, and fused them together to perform upsampling. Both colour features and depth features are exploited for segmentation with this FuseNet [31]. AdapNet was proposed by Valada et al. [32, 33] for semantic segmentation in adverse environments. A novel fusion technology called Convoluted Mixture of Deep Experts (CMoDE) was presented to enable a multi-stream network to learn features from different modalities. But this proposed network can only learn spatial features.

To the author's best knowledge, no one has used the temporal information for semantic pixel-wise segmentation along with CNNs so far. In this paper, we will investigate how to combine both spatial and temporal information for semantic pixel-wise segmentation.

Semantic Mapping with SLAM

For robotic applications, it is significant to locate the robot itself and percept the semantic environments simultaneously [34]. Salas-Moreno et al. [35] presented a planar SLAM system which could detect the planar in the environment and yield a planar map. A demonstration which replaced a wall with a Facebook web page was shown by the proposed system. They also proposed a SLAM system called SLAM++ [36]. The system detected objects such as chairs and desks and then utilised these objects for the localisation. However, only planar, desks, and chairs were extracted and perceived by the above SLAM system. More extensive semantic map could be built up for robotic applications by combining visual SLAM algorithms with the semantic segmentation. This is the research objective of this paper.

Approach

The proposed system is shown in Fig. 2. The details of the spatio-temporal neural network will be discussed first. Then, the geometry-based segmentation prediction for non-keyframes is followed. At last, we will present our method for 3D semantic mapping with visual SLAM technology.

Spatial Segmentation Network Architecture

The basic spatial neural network architecture we use in this paper is the PSPNet [4] which had an excellent performance in the ImageNet scene parsing challenge 2016. Its main advantage is the combination of very deep ResNet, dilated convolution, and pyramid pooling module.

At the very beginning, researchers preferred to use the standard convolution followed by pooling to extract the features and then adopted the deconvolution to recovery information from the feature maps. However, this method caused a loss to original details due to the use of pooling. It also needs to use intensive computational power and large memory. Chen et al. [15] proposed the dilated convolution which was also called atrous convolution. Its basic idea is to implement the convolution for feature maps with holes. By using the dilated convolution, the kernel of convolution is widened to some extent, the field-of-view is effectively enlarged, and the features are extracted and maintained efficiently. Further, it does not require any additional computation and memory. In order to explain the

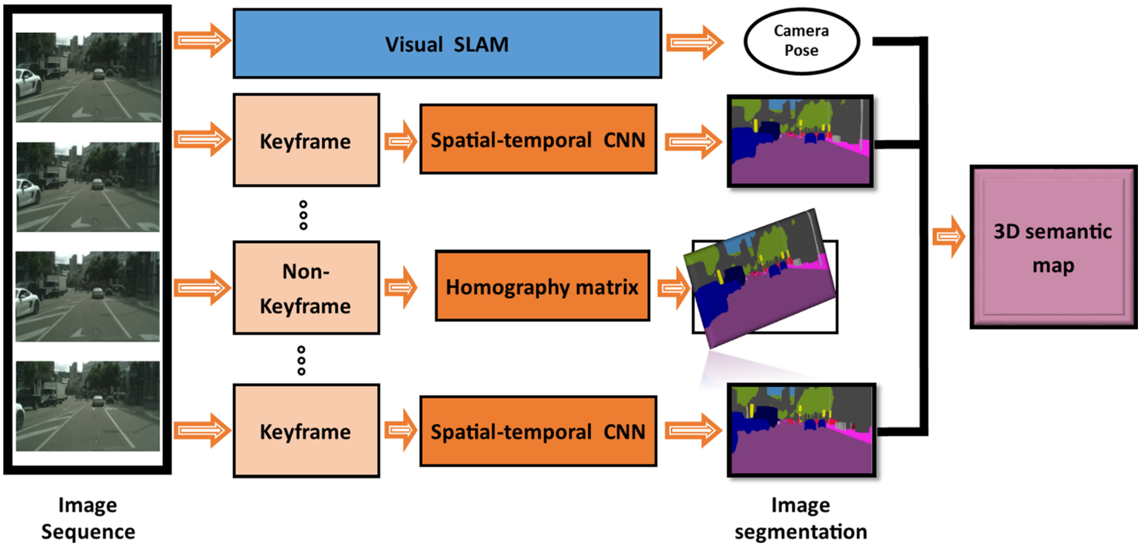

Fig. 2 System overview. For images from a video steam, we classify them as keyframes and non-keyframes according to the geometric constrains and time interval. For keyframes, we adopt the proposed spatio-temporal neural network to perform the semantic segmentation. For non-keyframes, we estimate the homograghy matrix (2D projective transformation) between the last keyframe and current frame. Then, the non-keyframe segmentation is predicted by the results from keyframes. At the same time, a visual SLAM algorithm is used for camera pose estimation. In the end, the 3D semantic map is constructed by our system

dilated convolution explicitly, we take a one-dimension signal, for example, $x[i]$ is the 1D input signal with length K, $y[i]$ is the output of the dilated convolution, so the dilated convolution can be defined as follows:

$$y[i] = \sum_{k=1}^{K} w[k] \cdot x[i + r \cdot k] \quad (1)$$

where rate r represents the stride that is used by the dilated convolution to sample the input signal $x[k]$. In this way, the dilated convolution has the functions which combine the standard convolution, pooling, and deconvolution. Compared with the traditional methods, it also enlarges the resolution of the resulting feature map while maintaining more information. In particular, the standard convolution is a special case of the dilated convolution where the rate r is 1.

The pyramid pooling module is another important factor that the PSPNet [4] outperforms other networks in semantic segmentation. As shown in Fig. 3, the pyramid pooling module is appended after the final feature map to better learn the contextual information. Four different pooling scales (1, 1/2, 1/3, 1/6) followed by convolutions are applied in this module. Then, four hierarchical feature maps are upsampled with bilinear interpolation. Finally, these learned features and the original feature map are concatenated into a new final global feature map to yield the segmentation result. In this way, the sub-region contextual information is better utilised along with the global contextual information.

Proposed Spatio-temporal CNN for Semantic Segmentation

Most existing neural networks take spatial images as the input and yield the semantic segmentation. In this section, we discuss how to use both spatial and temporal information for semantic pixel-wise segmentation. For dynamic video streams, long memory images (images recorded for a long time) play a less important role in current segmentation. It is hard to use long memory images to improve the segmentation accuracy on account of the fact that there is few pixel correspondences between the images. In contrast, short memory images are valued for semantic segmentation from video streams. Here, we use the image difference between two consecutive images as the temporal information. Then, we propose a CNN architecture, which has two streams, one is the colour image stream to capture the spatial features and another is the image difference stream to capture the temporal features, as shown in Fig. 3.

By applying convolution and softmax to the final feature maps after the pyramid pooling module, both spatial stream and temporal stream can generate the category prediction $P_s(x)$ and $P_t(x)$ for each pixel separately. We introduce three strategies to fuse the two streams together. (1) Pixel-wise prediction fusion—This fusion method is to treat the pixel-wise segmentation prediction from each stream as an independent normal distribution. Then, we can use the element-wise operation such as sum or max to fuse the prediction. (2) Feature map sum—This fusion method is to

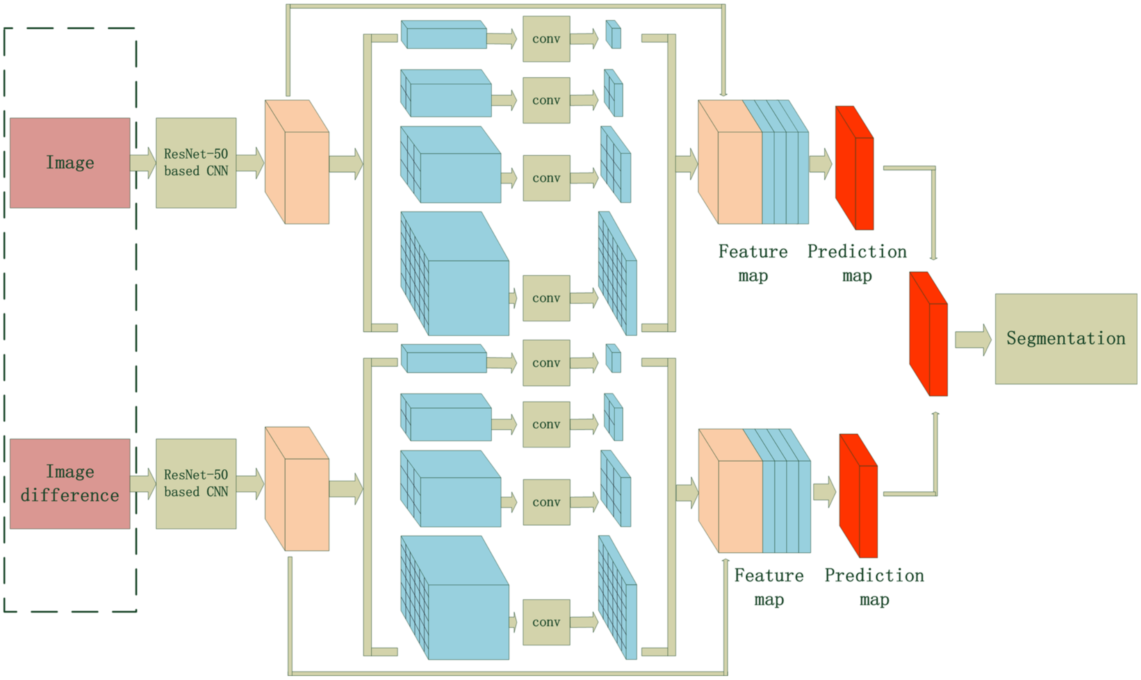

Fig. 3 Spatio-temporal neural network architecture. The colour image and the image difference are fed into the network. Then, the prediction maps are learned through the separated networks and fused together for image segmentation

implement the element operation for the final feature maps from the pyramid pooling module. The feature maps are added together in the element level, and then, we let the network to learn from the fused feature map. (3) Feature map concatenation—This fusion method is to concatenate the final feature maps from the two separate streams. By stacking them together into multiple channels, the neural network is trained end-to-end and the feature maps are fine-tuned. Detailed comparisons between these methods are given in IV-C (Fig. 4).

Image Segmentation Prediction with Homography Matrix

Pixel-level image segmentation with the network is time-consuming. We cannot use the network to segment every image for real-time applications. Due to the overlaps between consecutive images, we can just select some keyframes from the image sequence to perform the semantic segmentation with the proposed network. As shown in Fig. 2, we classify the images as keyframes and non-keyframes. For keyframe images, the pixel-level image segmentation is performed to yield the segmentation result. For non-keyframe image, the pixel-level segmentation map is predicted from the result of neighbour keyframe images. The segmentation prediction is conducted by using the homography matrix to predict the segmentation map of overlap regions. A 3 × 3 homography matrix H is computed first. It can map $[u, v]$, the 2D coordinate of a pixel in the keyframe, to $[u', v']$, the corresponding pixel in the non-keyframe. The matrix H is defined as below:

$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} H_{11} & H_{12} & H_{13} \\ H_{21} & H_{22} & H_{23} \\ H_{31} & H_{32} & H_{33} \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \tag{2}$$

The homography matrix can transform the segmentation map of a keyframe to the predicted segmentation map of the non-keyframe. The matrix $[H_{11}, H_{12}; H_{21}, H_{22}]$ is the rotation term, and the vector $[H_{13}, H_{23}]^T$ is the translation term. The rotation matrix, the translation vector, and the time interval jointly determine whether an image is a keyframe. Speciafically, we compare the norm of the rotation matrix, the translation vector, and the time interval with the corresponding thresholds. If one of these three is bigger than its corresponding threshold, we choose the frame as a keyframe. Only keyframes are processed to produce the semantic segmentation map. So the overall processing time is saved and the real-time performance is improved. Figure 5f shows the result of segmentation prediction for Fig. 5d. Although the segmentation performance of Fig. 5f is not as good as Fig. 5e, but it is still acceptable for robotic applications.

3D Semantic Scene Mapping with a SLAM Algorithm

The visual SLAM algorithm can simultaneously determine the robot pose and construct a 3D point cloud map for the environment.

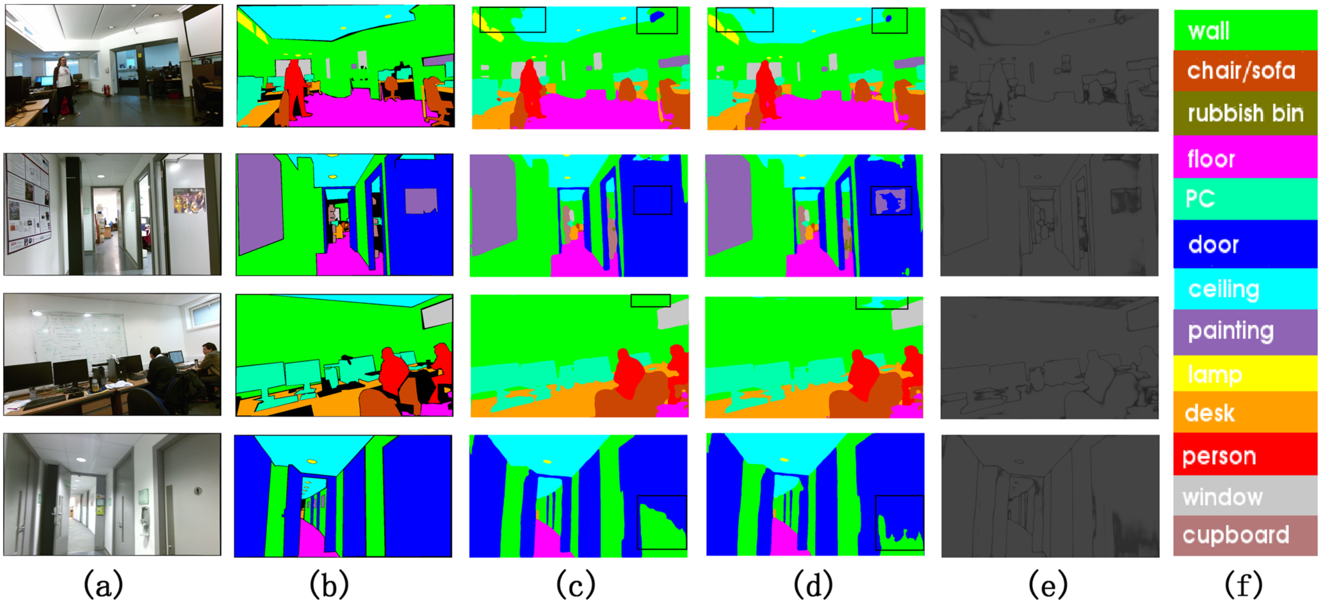

Fig. 4 Visual comparison on our manually labelled dataset (Essex Indoor). **a** Image. **b** Ground truth. **c** Spatial-PSPNet. **d** Spatio-temporal CNN. **e** ColorMap. **f** Uncertainty map for segmentation

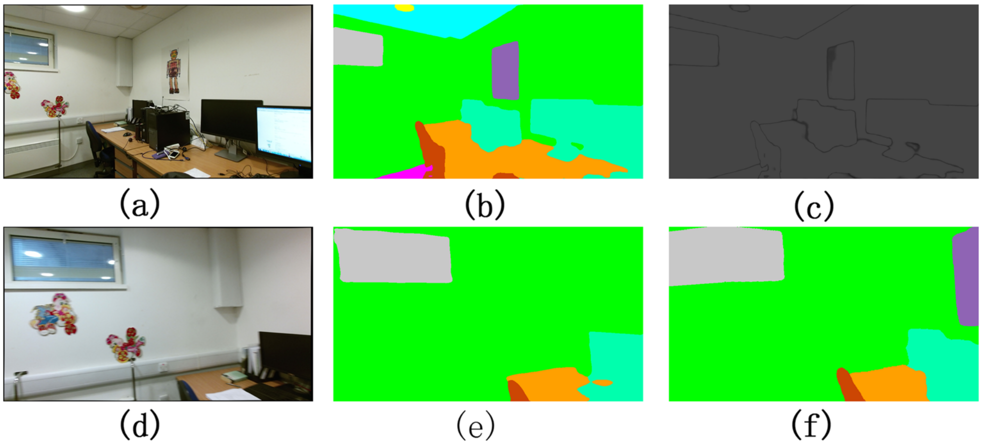

Fig. 5 Image segmentation prediction with homography matrix. **a** Keyframe. **b** Keyframe segmentation with CNN. **c** Uncertainty map for keyframe. **d** Non-keyframe. **e** Non-keyframe segmentation with CNN. **f** Segmentation prediction with homography matrix

But the 3D point cloud map is less meaningful for robotic applications. If each point in point cloud could be labelled with semantic meaning, a 3D semantic scene map is obtained, which is more meaningful for various robotic applications. That means we can simultaneously determine the robot pose and construct the 3D semantic scene map. In this paper, we combine the network for semantic segmentation with a visual SLAM algorithm to do this challenging task.

The system includes a spatial and temporal deep CNN proposed above and a standard visual SLAM algorithm. They run in parallel. The input to the SLAM algorithm is the images and the output is the robot pose and the 3D point cloud map. For each image, each point in the point cloud is labelled with the corresponding result in the semantic pixel-wise segmentation. Then, the next image is processed, and the labelled point cloud is merged together as a global 3D semantic map via the transformation of robot pose, i.e. the global semantic map can be obtained as below:

$$\text{Global Map}: \sum \mathbf{T}_{cw}\mathbf{X}_c = \sum_{i=1}^{n} \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{bmatrix} \mathbf{X}_c \qquad (3)$$

where $\mathbf{T}_{cw}$ is the 4×4 transformation matrix obtained from the visual SLAM algorithm. $\mathbf{T}_{cw}$ is composed of rotation $\mathbf{R}$ and translation $\mathbf{t}$ that can transform points from camera coordinate to world coordinate. $\mathbf{X}_c = (x_i, y_i, z_i, 1)^{\mathbf{T}}$ is the homogeneous position representation of a point in the camera coordinate.

Experimental Evaluation

In this section, we will evaluate the segmentation performance of our proposed spatio-temporal neural network and present a 3D semantic map system. We will first introduce our manually labelled dataset that was collected from our office. Secondly, we will discuss the fusion methods of spatial stream and temporal stream for segmentation prediction. Following that, the quantitative evaluations based on Cityscapes benchmark dataset will be given by comparing with different segmentation networks. Then, the qualitative evaluation will be presented. In the end, the 3D semantic map construction is demonstrated by our proposed system.

The proposed CNN is designed using the Caffe [37] platform, and all experiments are performed on a desktop equipped with one Nvidia GeForce Titan X GPU card and Intel Core i7-4790 4.0GHz CPU.

Indoor Dataset for Scene Segmentation

In this part, we introduce the dataset collected from our office environment in the network building of our university. In order to test the efficiency of temporal information for scene parsing, we need to have a dataset first. Most available datasets only contain discrete images and their ground truth labels. Video stream datasets for semantic segmentation are not readily available. So we have to use a Kinect One camera to build our own dataset for the second floor of our building. The scale of the second floor is about 40 m × 30 m. And the image size of our dataset is 960 × 540 in a pixel level. Both disparity image sequence and colour image sequence are provided. Then, we manually segment the collected images into 13 categories, as shown in Fig. 4. They are wall, chair, rubbish bin, floor, PC, floor, ceiling, painting, lamp, desk, person, and window. The semantic information of these categories is very important for robot navigation and robot-environment interface, especially the semantic information of floor, wall, ceiling, door, and person. In addition for training the network, our dataset is also used for 3D semantic scene map construction.

Training Details

We train the two streams separately first. The ResNet-50 [2]-based PSPNet [4] architecture is used for two separated streams. The model weights which have already been trained from the ImageNet scene parsing challenge 2016 are used for transfer learning for our dataset. Due to the limitation of GPU memory in our experiments, we choose the “cropsize” as 521 and the “batchsize” as 3. The “poly” learning rate policy is adopted for training. The base learning rate and the power are set to 0.00025 and 0.9, respectively. The weight decay and the momentum are set to 0.0001 and 0.9, respectively. The iteration number for training the two separated streams is 20,000. An auxiliary loss during training is used and the weight of this additional loss is set to 0.4. We also use the data augmentation during training. The image is randomly resized to 0.5 to 2, and the random mirror is also adopted.

For the dual-stream neural network training, we change the base learning rate to 0.0001. The “batchsize” is set to 1 because of memory limitation.

Fusion Method

In our proposed network, the image difference is used as the input of temporal stream. The image difference is the subtraction of current keyframe from the previous image. It can maintain the temporal information between two frames. We also tried to use optical flow in the experiments, but found that optical flow only represents moving objects and other information was lost. This leads to the poor performance in semantic segmentation.

In this part, we use our manually labelled dataset (Essex Indoor dataset) to test and compare different fusion methods. As explained in part III-B, we mainly compare three fusion methods here. As shown in Table 1, pixel-wise prediction sum is the best fusion method for semantic segmentation. The weight of different streams for fusion is an important parameter for segmentation accuracy. In the

Table 1 Segmentation performance comparison with different fusion methods

Method	Pixel-wise accuracy	Mean IoU
Spatial stream	88.91	95.93
Temporal stream	82.34	94.12
Feature map concatenation	89.21	96.30
Feature map SUM	89.85	96.46
Prediction MAX	89.46	96.46
Prediction SUM	*90.68*	*96.74*

The unit for pixel-wise accuracy and mean IoU is percentage (%)

The best results are shown in italic entries

Table 2 Per-class segmentation IoU (%) on Cityscapes dataset

Method	Road	Swalk	Build.	Wall	Fence	Pole	Tlight	Sign	Veg.	Terrain	Sky	Person	Rider	Car	Truck	Bus	Train	Mbike	Bike	mIoU
CRF-RNN [12]	96.3	73.9	88.2	47.6	41.3	35.2	49.5	59.7	90.6	66.1	93.5	70.4	34.7	90.1	39.2	57.5	55.4	43.9	54.6	62.5
FCN [3]	97.4	78.4	89.2	34.9	44.2	47.4	60.1	65.0	91.4	69.3	93.9	77.1	51.4	92.6	35.3	48.6	46.5	51.6	66.8	65.3
ParseNet [9]	97.5	78.5	89.5	40.4	45.9	51.1	56.8	65.3	91.5	69.4	94.5	77.5	54.2	92.5	44.5	53.4	49.9	52.1	64.8	66.8
DeepLab [16] (ResNet-101)	*97.9*	*81.3*	90.3	*48.8*	47.4	49.6	57.9	67.3	*91.9*	*69.4*	94.2	*79.8*	*59.8*	*93.7*	56.6	67.5	*57.5*	*57.7*	68.8	70.4
Spatial-PSPNet [4] (ResNet-50)	97.2	79.7	90.7	43.1	51.3	52.0	59.4	69.0	91.4	61.3	94.1	77.4	50.4	93.0	65.4	73.8	53.3	54.9	72.4	70.0
Spatio-temporal CNN	97.4	80.8	*91.2*	47.3	*52.9*	*56.2*	*60.8*	*71.5*	91.7	60.9	*94.2*	77.8	48.7	93.4	*66.8*	*75.1*	57.0	53.3	*73.4*	*71.1*

All methods are trained with the fine annotations of training dataset. Among these methods, CRF-RNN [12] and DeepLab [16] use CRFs as the post-processing

The best results are shown in italic entries

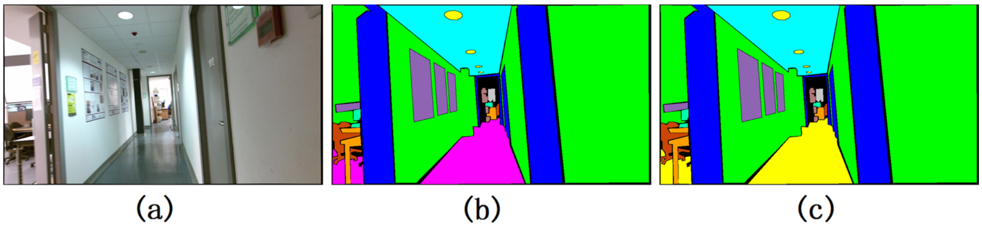

Fig. 6 Train the network with the “problem” dataset. We deliberately labelled the floor in an image as the lamp in the ground truth image. **a** The image in the training dataset. **b** The ground truth with right label in the training dataset. **c** The ground truth with wrong label in “problem” dataset

experiment, the weight of spatial stream is set to 0.7, and the weight for temporal stream is set to 0.3. The stage-by-stage network training mechanism is used for our spatio-temporal CNN. We first train the spatial stream and the temporal stream separately. Then, we transfer the learned network weights from the separate single streams to the dual-stream CNN and finally fine-tune the dual-stream CNN.

Quantitative Analysis

In this part, we compare the proposed spatio-temporal neural network with other CNNs for semantic segmentation. The public dataset Cityscapes [20] is taken as the benchmark. The images of Cityscapes dataset are collected in the urban environment from different cities, and the camera is fixed in the car.

Considering potential robotic applications, we only use the PC with one graphic card for processing. This means the memory is limited for training when compared with the PSPNet [4]. The PSPNet adopted Rsenet-101 [2] as its basic network architecture for Cityscapes dataset. They used four GPUs for training and have much more memory. So when training the spatial stream and temporal stream in our experiments, we adopt the ResNet-50-based PSPNet and set the “cropsize” and “batchsize” to 617 and 2, respectively. For the dual-stream CNN training, we set the “cropsize” and “batchsize” to 569 and 1, respectively. The spatio-temporal CNN is much bigger than the single streams and needs more memory for training. The transfer learning is used first, then we fine-tune the network weights from the model learned from ADK20K [22] dataset.

The results are showed in Table 2. Deeplab [16] uses CRFs as a post-processing method to enhance the performance. The proposed spatio-temporal CNN does not use CRFs, but outperforms the Resnet-50-based spatial-PSPNet and other networks. Figure 9 lists some results, and we can see that the segmentation performance is improved by using the temporal information.

Qualitative Analysis

This part evaluates the benefit of using the temporal information for CNN to process video streams. Among the training dataset, we deliberately label one category in one image wrongly while all other images are labelled correctly. If the network is robust to the problem, it should be able to distinguish the wrong label. As shown in Fig. 6, we deliberately label the floor (bright purple) as the lamp (yellow) in one training image. Then, the “problem” dataset is fed to the network for training.

Table 3 shows the mIoU results on the Essex Indoor dataset. All the networks are trained with the “problem” dataset. As shown in the table, the spatio-temporal CNN demonstrates the best performance in semantic scene segmentation.

Figure 7 shows the segmentation results for different neural networks. Figure 7a is selected from the test dataset. It is the neighbour image of Fig. 6a. These two images have some similarities but are totally different. Figure 7d is the segmentation result from the spatial stream with colour images as the input. Figure 7e is the segmentation result from the temporal stream with image difference as

Table 3 Per-class segmentation IoU (%) on the Essex Indoor test dataset when trained with the “problem” dataset

Method	Wall	Ceiling	Floor	Lamp	Desk	Person	Chair	PC	Door	Painting	Windows	Cupboard	Rubbish Bin	mIoU
Spatial-PSPNet	90.27	89.21	98.95	70.30	95.30	87.19	89.69	91.17	81.26	79.90	88.22	70.27	97.76	86.88
Temporal-PSPNet	89.92	87.05	90.49	57.24	75.55	83.26	77.35	79.85	81.39	84.62	87.99	73.30	93.76	81.67
Spatio-temporal CNN	95.45	89.97	96.26	64.18	91.42	94.72	88.33	93.33	93.74	90.10	93.10	83.26	98.83	*90.21*

The best results are shown in italic entries

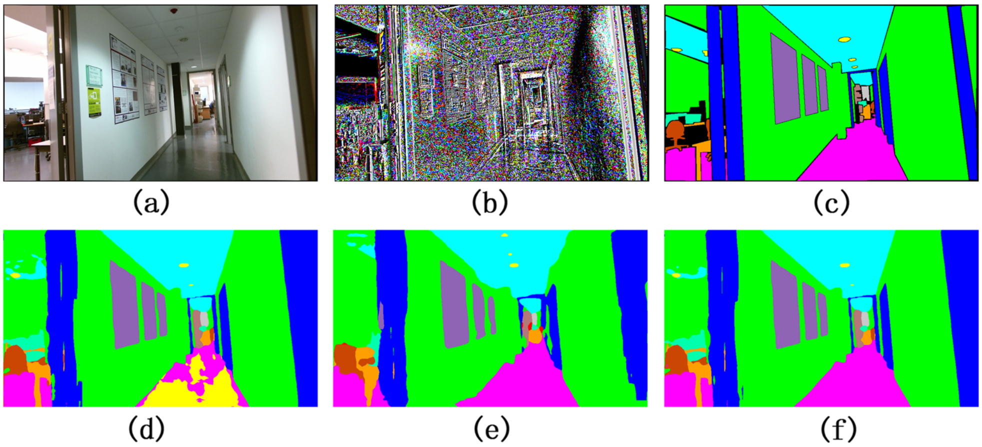

Fig. 7 Segmentation results of the networks using "problem" dataset for training. **a** Colour image in the test dataset. **b** Difference image in the test dataset. **c** Ground truth. **d** Segmentation result with colour image as inputs. **e** Segmentation result with image difference as inputs. **f** segmentation result with both colour image and image difference as inputs

the input. Figure 7e is the segmentation result from our proposed spatio-temporal CNN. As shown in the figure, the spatial stream segments the floor as the lamp, i.e. it fails to segments the floor, while the proposed fusion CNN segments the floor successfully. The floors in other test images are all segmented successfully with the spatial stream. This result means only using colour images as the network input cannot find the "problem" in the dataset, while using both spatial and temporal information can make right segmentation decision when facing with the "problem" dataset.

3D Semantic Mapping

A 3D semantic map is very useful for robotic applications. By constructing the 3D semantic map, the robots can interact autonomously with the environment. For example, the robots can navigate themselves in an unknown environment by detecting the road and the robots can implement grasping tasks by detecting different objects in the 3D space.

We obtain the 3D semantic map of our second floor in the building by combining the spatio-temporal neural network with a SLAM algorithm. The state-of-the-art SLAM algorithm (ORB-SLAM [38, 39]) is used, which is able to obtain the camera pose and the point cloud map in real time. The keyframes are selected and then fed to the spatio-temporal CNN for semantic segmentation. Compared with the system without keyframe structure, the real-time performance speeds up from 3 to 11 Hz. By labelling the point cloud using the result of semantic pixel-wise segmentation, the 3D semantic map is constructed and shown in Fig. 8. Although there is some noise points in the map, it does provide the meaningful information for potential robotic applications. The main cause for the noise points is due to the measurement limitation of the depth camera. We plan to tackle the problem in our future work.

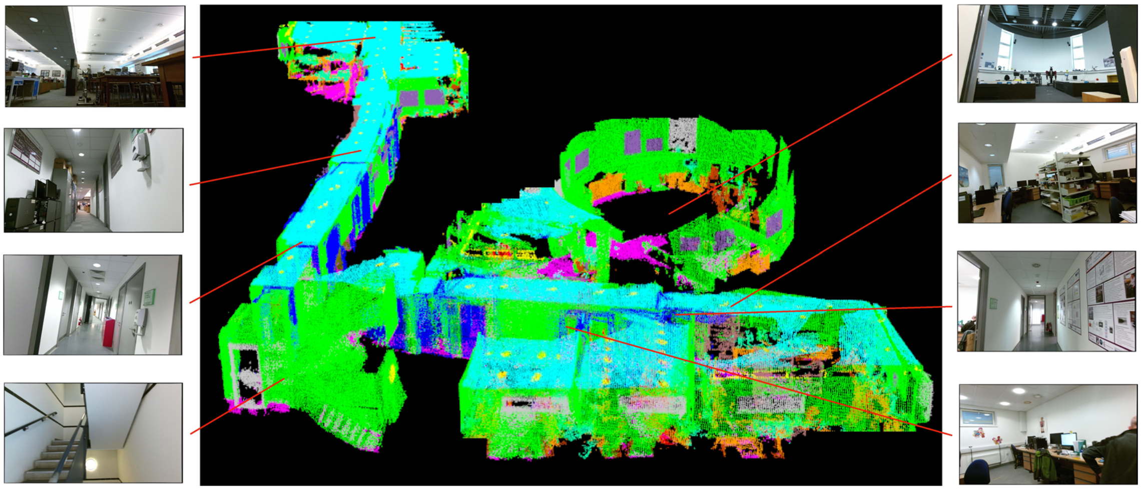

Fig. 8 3D semantic map of the second floor in the network building of our university

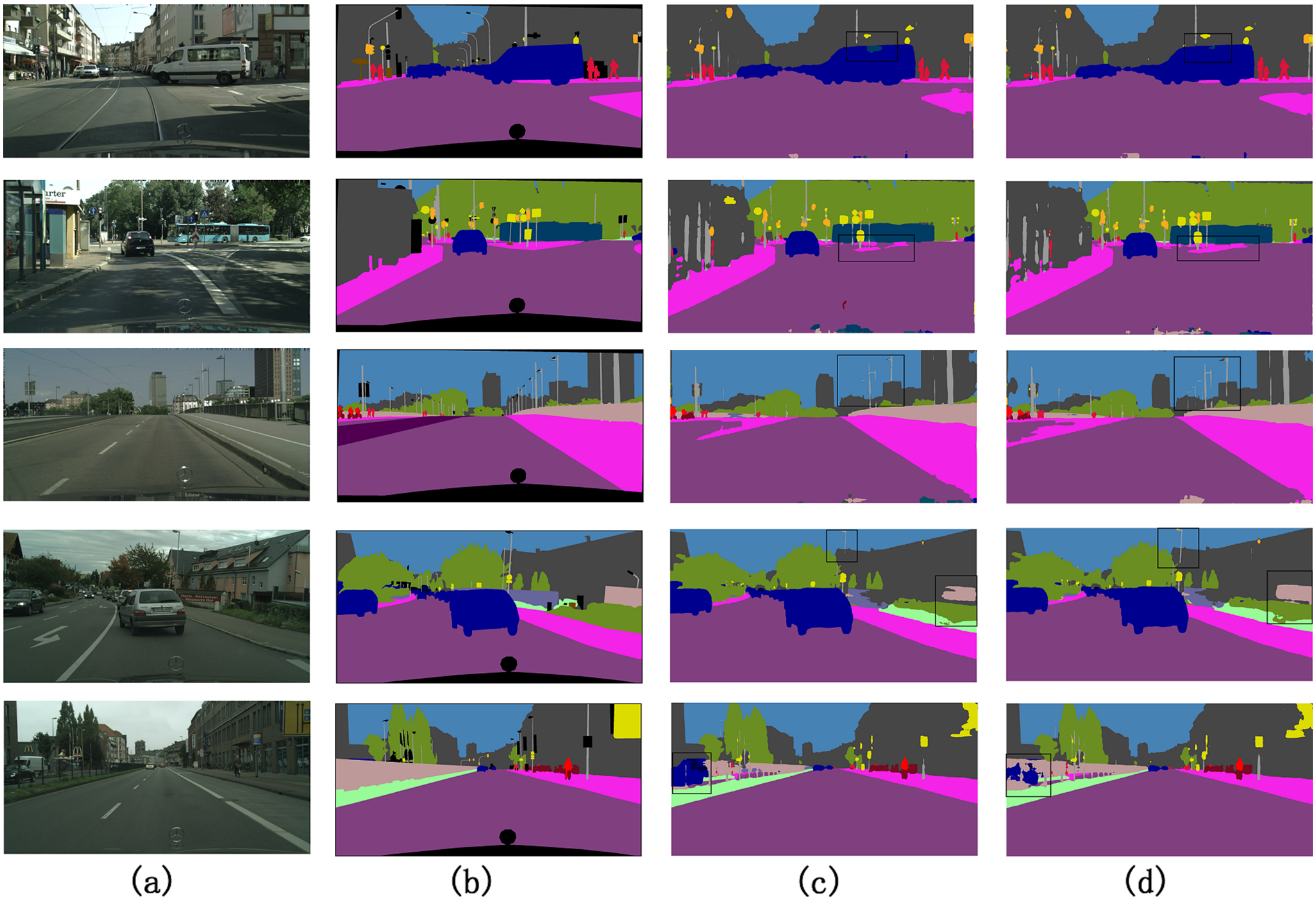

Fig. 9 Visual comparison on Cityscapes dataset. **a** Image. **b** Ground truth. **c** Spatial-PSPNet. **d** Spatio-temporal CNN

Conclusions

In this paper, we have presented a novel spatio-temporal CNN for image segmentation which shows a better performance when compared with the CNNs using only spatial information (Fig. 9). The image difference is taken as the temporal information for additional network input in the proposed network. Different fusion methods for spatial and temporal information are discussed and compared. A global 3D semantic map is constructed with the proposed system which combines the spatio-temporal CNN with a SLAM algorithm. However, there are some noisy points in the constructed 3D map. This is caused by the limitation of depth camera and the wrong segmentation of the scene in some images. In the future, we would like to investigate how to improve the constructed 3D map.

Acknowledgments The authors would like to thank Robin Dowling for his support in experiments.

Funding Information The first author has been financially supported by scholarship from China Scholarship Council.

Compliance with Ethical Standards

Conflict of Interest The authors declare that they have no conflict of interest.

Informed Consent Informed consent was obtained from all individual participants included in the study.

Human and Animal Rights This article does not contain any studies with human or animal subjects performed by the any of the authors.

References

1. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A. Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2015. p. 1–9.
2. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2016. p. 770–8.
3. Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2015. p. 3431–40.

4. Zhao H, Shi J, Qi X, Wang X, Jia J. Pyramid scene parsing network. arXiv:1612.01105. 2016.
5. Durrant-Whyte H, Bailey T. Simultaneous localization and mapping: part I. IEEE Robot Autom Mag. 2006;13(2):99–110.
6. Bailey T, Durrant-Whyte H. Simultaneous localization and mapping: part II. IEEE Robot Autom Mag. 2006;13(3):108–17.
7. Xie J, Yu L, Zhu L, Chen X. Semantic image segmentation method with multiple adjacency trees and multiscale features. Cogn Comput. 2017;9(2):168–79.
8. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. In: Proceedings of the 3rd international conference on learning representations; 2015. p. 1–14.
9. Liu W, Rabinovich A, Berg AC. Parsenet: looking wider to see better. arXiv:1506.04579. 2015.
10. Badrinarayanan V, Kendall A, Cipolla R. Segnet: a deep convolutional encoder-decoder architecture for image segmentation. arXiv:1511.00561. 2015.
11. Kendall A, Badrinarayanan V, Cipolla R. Bayesian segnet: model uncertainty in deep convolutional encoder-decoder architectures for scene understanding. arXiv:1511.02680. 2015.
12. Zheng S, Jayasumana S, Romera-Paredes B, Vineet V, Su Z, Du D, Huang C, Torr PH. Conditional random fields as recurrent neural networks. In: Proceedings of the IEEE international conference on computer vision; 2015. p. 1529–37.
13. Arnab A, Jayasumana S, Zheng S, Torr PH. Higher order conditional random fields in deep neural networks. In: European conference on computer vision. Springer; 2016. p. 524–40.
14. Deng J, Dong W, Socher R, Li L-J, Li K, Fei-Fei L. Imagenet: a large-scale hierarchical image database. In: IEEE conference on computer vision and pattern recognition, 2009. CVPR 2009. IEEE; 2009. p. 248–55.
15. Chen L-C, Papandreou G, Kokkinos I, Murphy K, Yuille AL. Semantic image segmentation with deep convolutional nets and fully connected CRFs. arXiv:1412.7062. 2014.
16. Chen L-C, Papandreou G, Kokkinos I, Murphy K, Yuille AL. Deeplab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. arXiv:1606.00915. 2016.
17. Chen L-C, Yang Y, Wang J, Xu W, Yuille AL. Attention to scale: scale-aware semantic image segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2016. p. 3640–9.
18. Everingham M, Eslami SA, Van Gool L, Williams CK, Winn J, Zisserman A. The pascal visual object classes challenge: a retrospective. Int J Comput Vis. 2015;111(1):98–136.
19. Wu Z, Shen C, Hengel AVD. High-performance semantic segmentation using very deep fully convolutional networks. arXiv:1604.04339. 2016.
20. Cordts M, Omran M, Ramos S, Rehfeld T, Enzweiler M, Benenson R, Franke U, Roth S, Schiele B. The cityscapes dataset for semantic urban scene understanding. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2016. p. 3213–23.
21. Wu Z, Shen C, Hengel AVD. Wider or deeper: revisiting the resnet model for visual recognition. arXiv:1611.10080. 2016.
22. Zhou B, Zhao H, Puig X, Fidler S, Barriuso A, Torralba A. Semantic understanding of scenes through the ade20k dataset. arXiv:1608.05442. 2016.
23. Tu Z, Abel A, Zhang L, Luo B, Hussain A. A new spatio-temporal saliency-based video object segmentation. Cogn Comput. 2016;8(4):629–647.
24. Doborjeh ZG, Doborjeh MG, Kasabov N. Attentional bias pattern recognition in spiking neural networks from spatio-temporal EEG data. Cogn Comput, 2017:1–14.
25. Wang S, Clark R, Wen H, Trigoni N. DeepVO: towards end-to-end visual odometry with deep recurrent convolutional neural networks. In: 2017 IEEE international conference on robotics and automation (ICRA). IEEE; 2017. p. 2043–50.
26. Wang L, Xiong Y, Wang Z, Qiao Y. Towards good practices for very deep two-stream convnets. arXiv:1507.02159. 2015.
27. Wang L, Xiong Y, Wang Z, Qiao Y, Lin D, Tang X, Van Gool L. Temporal segment networks: towards good practices for deep action recognition. In: European conference on computer vision. Springer; 2016. p. 20–36.
28. Li R, Liu Q, Gui J, Gu D, Hu H. Indoor relocalization in challenging environments with dual-stream convolutional neural networks. IEEE Trans Autom Sci Eng. 2017.
29. Eitel A, Springenberg JT, Spinello L, Riedmiller M, Burgard W. Multimodal deep learning for robust RGB-d object recognition. In: 2015 IEEE/RSJ international conference on intelligent robots and systems (IROS). IEEE; 2015. p. 681–7.
30. Schwarz M, Schulz H, Behnke S. RGB-D object recognition and pose estimation based on pre-trained convolutional neural network features. In: 2015 IEEE international conference on robotics and automation (ICRA). IEEE; 2015. p. 1329–35.
31. Hazirbas C, Ma L, Domokos C, Cremers D. Fusenet: incorporating depth into semantic segmentation via fusion-based CNN architecture. In: Proceedings of ACCV; 2016.
32. Valada A, Oliveira G, Brox T, Burgard W. Towards robust semantic segmentation using deep fusion. In: Robotics: science and systems (RSS 2016) workshop, are the sceptics right? Limits and potentials of deep learning in robotics; 2016.
33. Valada A, Vertens J, Dhall A, Burgard W. Adapnet: adaptive semantic segmentation in adverse environmental conditions. In: 2017 IEEE international conference on robotics and automation (ICRA). IEEE; 2017.
34. Hülse M, McBride S, Lee M. Fast learning mapping schemes for robotic hand–eye coordination. Cogn Comput. 2010;2(1):1–16.
35. Salas-Moreno RF, Glocken B, Kelly PH, Davison AJ. Dense planar slam. In: 2014 IEEE international symposium on mixed and augmented reality (ISMAR). IEEE; 2014. p. 157–64.
36. Salas-Moreno RF, Newcombe RA, Strasdat H, Kelly PH, Davison AJ. Slam++: simultaneous localisation and mapping at the level of objects. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2013. p. 1352–9.
37. Jia Y, Shelhamer E, Donahue J, Karayev S, Long J, Girshick R, Guadarrama S, Darrell T. Caffe: convolutional architecture for fast feature embedding. In: Proceedings of the ACM international conference on multimedia. ACM; 2014. p. 675–8.
38. Mur-Artal R, Tardós JD. Fast relocalisation and loop closing in keyframe-based SLAM. In: 2014 IEEE international conference on robotics and automation (ICRA). IEEE; 2014. p. 846–53.
39. Mur-Artal R, Montiel J, Tardos JD. ORB-SLAM: a versatile and accurate monocular SLAM system. IEEE Trans Robot. 2015;31(5):1147–63.