Spatio-Temporal Road Scene Reconstruction using Superpixel MRF

Yaochen Li, Member, IEEE, Yuehu Liu, Member, IEEE, Jihua Zhu, Shiqi Ma, Zhenning Niu, and Rui Guo

Abstract—Scene models construction based on image rendering is a hot topic in the computer vision community. In this paper, we propose a framework to construct road scene models based on 3D corridor structures. The construction of scene models consists of two successive stages: road detection and scene construction. The road detection is implemented via a new superpixel Markov random field (MRF) algorithm. The data fidelity term of the energy function is jointly computed using the superpixel features of color, texture, and location. The smoothness term is defined by the interaction of spatio-temporally adjacent superpixels. The control points of road boundaries are generated with the constraint of vanishing point. Subsequently, the road scene models are constructed, where the foreground and background regions are modeled independently. Numerous applications are developed based on the proposed framework, e.g., traffic scenes simulation. The experiments and comparisons are conducted for both the road detection and scene construction stages, which prove the effectiveness of the proposed method.

Index Terms—Superpixel, Markov random field, region detection, scene modeling.

I. INTRODUCTION

WITH the rapid development of multimedia technology, scene construction based on image rendering becomes important to many applications. The scene construction results can be applied to simulate the events of the real world. The typical applications are threefold. (1) Virtual street view [1][2]. Road image sequences contain lots of useful information for people’s lives. The Google Company sent a fleet of cars to capture the road images all over the world. The Google Street View system was developed to allow users tour into the street scenes while browsing the map. Another similar application is Microsoft Street Slide, which introduces the scene bubbles to interact with the users. (2) Intelligent transportation systems [3][4]. The advanced driver assistant systems (ADAS) are promoted by the virtual scene reconstruction. In contrast to the traditional field test of unmanned vehicles, the evaluation can be implemented in the virtual environment. The off-line test for unmanned vehicles is much safer and saves time and energy. (3) Free viewpoint television [5][6]. The virtual reality technology is commonly used in the modern film industry. The free viewpoint video becomes popular, for it provides users with more interactions. Users can select their viewpoint freely while browsing the videos by the free viewpoint rendering methods.

In early studies, the virtual scenes are constructed by computer graphics methods. The typical examples include the Prescan software of TNC [7] and the CarMaker software of IPG Company [7]. Besides the method of computer graphics, the road scene construction from image sequences becomes popular. The Waymo team of Google Company propose a virtual platform for scene modeling, utilizing machine learning methods from road images. The off-line test of unmanned vehicles is conducted in the platform. Recently, the generative adversarial networks (GAN) is proposed for the synthesis of virtual traffic scenes [8]. However, these methods have many limitations, such as the modeling of both foreground and background, the rendering of new viewpoint images, etc.

The vision system of “component generation-depth computation-3D recovery” was considered a classical model in computer vision community. Besides the reconstruction of semantic objects, the semantic analysis and modeling of the entire scene becomes an important research problem. In order to implement scene analysis and modeling, the region detection and semantic reasoning are crucially important. In this paper, a new framework to model 3D scenes is introduced using David Marr’s theory [9]. The road regions of the input image sequences are detected via a new superpixel-based Markov random field (MRF) method. The energy function is defined based on vision features computed in spatio-temporal domains. The energy minimization process follows a cycle of “global energy initialization-local energy comparison-global energy comparison”. After the detection of road regions, the 3D corridor-structured scene models are constructed based on the control points of road boundaries. The road regions are assumed to be the horizontal plane, while the rest of the scene components stand perpendicularly to the road plane. The panoramic scene models can be constructed as well, and useful applications are developed, such as virtual street view, traffic scene simulation, etc.

In summary, the main contributions of this paper are threefold:

- A new superpixel-based MRF method to detect road regions in image sequences. The data fidelity term of the energy function is defined by the combination of
superpixel features of color, texture and location. The smoothness term is computed by the feature distance of spatio-temporally adjacent superpixels. The energy minimization is implemented via a cycle of “global energy initialization-local energy comparison-global energy comparison”.

- A novel framework to construct road scene models automatically from image sequences. The road scene model follows the 3D corridor structure, where the road regions are assumed to be the horizontal planes of the scene models. The foreground objects are modeled independently of the background models. Moreover, the panoramic scene models can be constructed.
- A new multimedia system for the interactive tour in traffic scenes based on the scene models. Two basic modes are designed: bird-view mode and touring mode. New traffic elements can be inserted into the traffic scenes, which can be applied to the virtual simulation of unmanned vehicles.

II. RELATED WORKS

The research work of this paper basically includes the road detection based on superpixel MRF, as well as scene models reconstruction. We introduce the related research for these components, respectively:

1) Region detection: The semantic analysis of road images is important to the 3D road scene reconstruction, while road detection is the basis for such semantic analysis task. Belaid et al. [13] propose a watershed transformation method, which can be applied to road region detection. The topological gradient method is utilized to avoid the over segmentation. Jiang et al. [15] propose a method to extract spatio-temporally consistent segments from a video. The corresponding depth data is estimated and 3D reconstruction is implemented. Song et al. [10] propose a new image appearance transfer method combining color and texture features. The feature detection and matching between source and reference images are implemented. The image and geometric features can be exploited. Peng et al. [10] propose a method to extract superpixels using a higher order optimization framework. The K-means clustering technique is adopted, and a higher order energy function is employed to optimize the initial superpixels. The higher order energy method is also used in the study of [17]. Gould et al. [19] present a semantic segmentation method by label transfer. The approximate nearest neighbor algorithm is applied to build a graph over superpixels. Although it’s effective in the standard datasets, the experiments on road image sequences are not always satisfying. Lu et al. [20] select superpixel-level seeds in an unsupervised way, and then find the appropriate superpixel neighbors according to the GrowCut framework. However, the vanishing points should be first detected as a supplement.

2) Superpixel MRF: The pixel-level MRF algorithms are widely used for image segmentation and annotation. Kim et al. [22] apply MRF model to image segmentation, which is effective for outdoor images. Yang et al. [23] propose a pixons-based MRF method for image segmentation. Less computation cost of the energy optimization is demanded than the traditional MRF algorithms. However, the generation of pixons requires the solving of anisotropic diffusion equation. Elia et al. [24] propose a tree-structured MRF model, which can be applied to Bayesian image segmentation. The input image can be iteratively segmented into smaller regions based on a “split-merge” step. Numerous approaches have been developed to improve the speed of MRF. Schick et al. [25] propose a superpixel-based MRF algorithm, which applies a classical Graphcut method to energy minimization. However, the data fidelity term is too simple to concern about the irregular attributes of superpixels. Pei et al. [26] propose a belief propagation method to detect the image regions based on superpixel saliency. The vision features of superpixels are then extracted to optimize the saliency regions. However, the strong contrast between the foreground and background regions is necessary. The methods proposed in [27] also have the drawbacks of insufficient use of temporal information, complex iteration steps of energy function and low efficiency of data fidelity terms. To overcome the above drawbacks and limitations, we propose a new superpixel MRF algorithm to detect the road regions of images and videos. In our method, the data fidelity term of the energy function combines the various types of superpixel features. The smoothness terms are computed based on spatial-temporal neighborhood superpixels.

3) Scene models construction: 3D modeling and reconstruction is currently a hot topic in the communities of computer vision and computer graphics. For the reconstruction of single objects, important achievements have been reached for 3D building reconstruction, 3D models retrieval, 3D face simulation, etc. These technologies aim to recover the 3D geometric surface of single objects. Besides the single objects reconstruction, the 3D reconstruction of the entire scene requires more challenging technologies. The framework of Tour into the picture (TIP) proposed by Horry et al. [11] aims to recover the 3D structure from a single still image. After the detection of the vanishing point, the input image is roughly partitioned into “left wall”, “right wall”, “back wall” and “road plane”. The foreground models can be constructed independently of the background models. However, the TIP model is not fit for the curved-edge road conditions. Saxena et al. [12] learn plane parameters based on MRF, which are utilized to judge the locations and orientations of the recovered mesh facets. The Make3D models with wireframe meshes are then constructed. The monocular image features utilized mainly include color, texture, gradient and edges. However, this method makes no prior assumption for large-scale semantic environment. Hoiem et al. [13] perform superpixel segmentation for input images, and then apply support vector machine (SVM) to cluster the superpixels of similar appearance into cliques. The region properties of the superpixel cliques are then specified to construct the “pop up” style 3D scene models. Lou et al. [14] first predict the global image structure based on layout templates, and then use random walks to infer the refined scene structures. However, the limited number of scene stages cannot be applied to all the road scene structures.

The rest of the paper is summarized as follows: In Section III, the MRF model at pixel level is introduced. The region detection method using superpixel MRF is described in Section
IV. Spatio-Temporal Road Region Detection

Motivated by the pixel-level MRF model, we propose a new MRF model based on superpixels. The superpixel segmentation for each input image should be first implemented for the proposed method.

A. Superpixels and Feature Pools

Thousands of superpixels may exist in a single still image, which are considered as the mid-level image units. Useful image features can be generated by clustering the superpixels into cliques. A superpixel clique can occupy an image region that represents a certain semantic object in an ideal condition. However, the clique may not correspond to a semantic object even though all the superpixels have the same label. The superpixel features are widely used in the vision tasks of image segmentation, semantic annotation, etc. These features mainly include color, texture, and geometric shapes, as shown in Table I.

| Feature Descriptors                  | Feature Numbers |
|--------------------------------------|-----------------|
| Color                                | 9               |
| C1 RGB color                         | 3               |
| C2 HSV color                         | 3               |
| C3 CIELAB                            | 3               |
| Texture                              | 62              |
| T1 Gabor filters: 4 scales, 6 orientations | 48            |
| T2 Local binary pattern: 3 × 3 template | 9           |
| T3 Edge histogram descriptors        | 5               |
| Locations and Shapes                 | 6               |
| L1 Location: Normalized x and y coordinates | 2           |
| L2 Shapes: Superpixel number in the clique | 1           |
| L3 Shapes: Edge number within convex hull | 1           |
| L4 Shapes: Ratio of the pixels to the convex hull | 1 |
| L5 Shapes: Whether the clique is continuous | 1          |

After the extraction of the superpixel features, the feature pool for each image region can be constructed. The color feature pools are built based on the K-means clustering of CIELab color space. The transformation from RGB to CIELab color space can be easily implemented. For each superpixel belonging to a certain image region, its mean and variance in the CIELab space are computed. The color feature pools are then constructed by the clustering of the color means.

To represent the texture features, the local texture is represented by the intensity distributions of adjacent pixels. The global texture is defined by the duplicate of local texture information. The texture feature pools are constructed based on the clustering of Gabor filter outputs for all the image regions [28]. For each pixel in the image, the corresponding output vector of Gabor filter is computed. We choose the Gabor filter with 4 scales and 6 directions for implementation. For example, the output vector dimension of the Gabor filter with $128 \times 128$ size is 393216. The computation for such high-dimensional feature vector is very time-consuming. We only choose the mean and variance of the Gabor filter output for improvement. As a result, the feature dimension is only 48 for the aforementioned Gabor filter.

III. MRF Model at Pixel Level

First of all, we introduce the MRF model at pixel level. Two groups of random variables $X$ and $Y$ are defined, which refer to the observation data and state variables, respectively. The posterior probability distribution $P(Y|X)$ can be solved based on the Bayesian theory [21]:

$$P(Y|X) = \frac{P(X|Y)P(Y)}{P(X)}, \quad (1)$$

where $P(Y)$ is the prior probability of $Y$, $P(X|Y)$ is the conditional probability distribution given $Y$, $P(X)$ is the density function of $X$, which is a constant when $X$ is identified. $y_i$ and $y_j$ denote the labels of the pixels $i$ and $j$.

For the task of image annotation, $X$ denotes the pixel coordinate set in the image, while $Y$ denotes the corresponding label set. The constraint between these variables is established to compute $P(Y)$, which is assumed to satisfy the uniform distribution:

$$P(Y) = \frac{1}{Z} \exp\left(-\sum_{i,j \in N_i} f_{ij}(y_i, y_j)\right), \quad (2)$$

where $f_{ij}(\bullet)$ denotes the uniform Gibbs distribution. The neighborhood domain of $i$ is denoted by $N_i$. $Z$ is used for normalization.

$P(X|Y)$ is the likelihood function. Assuming the elements in $X$ and $Y$ are independent of each other:

$$P(X|Y) = \prod_i f_i(x_i, y_i) = \exp\left(\sum_i \log f_i(x_i, y_i)\right), \quad (3)$$

where $f_i(\bullet)$ is the data fidelity term of the $i$th element.

Based on the above analysis, Eq.(4) can be inferred by:

$$P(X|Y)P(Y) = \frac{1}{Z} \exp\left(\sum_i \log f_i(x_i, y_i)\right) \exp\left(-\sum_{i,j \in N_i} f_{ij}(y_i, y_j)\right). \quad (4)$$

When the constant component is removed, the energy function is defined as follows:

$$E(Y|X) = \sum_{i,j \in N_i} f_{ij}(y_i, y_j) - \sum_i \log f_i(x_i, y_i). \quad (5)$$

The variable $Y$ is solved by the minimization of energy function:

$$Y^* = \arg\min_Y E(Y|X). \quad (6)$$

The classical algorithms can be applied to the energy minimization, such as greedy algorithm, dynamic programming algorithm, etc. The pixel-level MRF is useful for the tasks of image segmentation, saliency detection, etc. However, the pixel-level MRF model has two main disadvantages: (1) too much time cost and (2) low dimension of vision feature vectors.

IV. Spatio-Temporal Scene Reconstruction Process

Experiments and discussions are conducted in Section VI. Finally, we close this paper with the conclusion and future works.
B. Road Region Detection

After the construction of feature pools, we then propose the road region detection algorithm via superpixel MRF. The selection of the superpixel neighborhoods is mainly based on the adjacency relationship. The connections of spatial and spatio-temporal adjacent superpixels are shown in Fig. 2 (a) and (b). The red dots denote the current superpixel for processing. The green dots refer to the spatially adjacent superpixels, while the yellow dots refer to superpixels in both spatial and temporal domains.

Fig. 2 shows the flow diagram of road detection process, which is fit for both single image and image sequences. We mainly introduce the algorithm for image sequences. In our algorithm, the global energy based on superpixel MRF is defined as follows:

$$E_i = - \sum_{i \in S} f_i(x_i|y_i) + \sum_{i \in S} \sum_{j \in N_{spa}(i)} \lambda_1 f_{ij}(y_i, y_j) + \sum_{i \in S} \sum_{j \in N_{tem}(i)} \lambda_2 f_{ij}(y_i, y_j)$$

where $S$ denotes the superpixel set. $x_i$ is the appearance feature of the $i_{th}$ superpixel. $y_i$ denotes the label of the $i_{th}$ superpixel, which is in the set of $\mathcal{L} = \{0, 1\}$. $\log p(x_i|y_i)$ denotes the data fidelity term of the $i_{th}$ superpixel. $f_{ij}(\cdot)$ is the smoothness term of the adjacent superpixel pairs. $N_{spa}(i)$ and $N_{tem}(i)$ denote the adjacent neighbors of superpixel $i$ in spatial and temporal domains, respectively. $\lambda_1$ and $\lambda_2$ are constant values.

The probability of data fidelity term is computed by the combination of color, texture and location:

$$p(x_i|y_i) = p(c_i|y_i) p(t_i|y_i) p(h_i|y_i),$$

where $c_i$, $t_i$ and $h_i$ denote the color, texture and location features of the $i_{th}$ superpixel, respectively.

The color probability is defined by the three-channel Gaussian distribution:

$$p(c_i|y_i) = \sup_{(\mu_m, \Sigma_m) \in \mathbb{C}} \left\{ \frac{1}{(2\pi)^{d/2} |\Sigma_m|^{1/2}} \exp\left(-\frac{1}{2}(c_i - \mu_m)^T \Sigma_m^{-1} (c_i - \mu_m)\right) \right\},$$

where $\mu_m$ and $\Sigma_m$ are the $m_{th}$ mean and covariance in the color feature pool corresponding to label $y_i$.

The road region usually contains many texture materials and stripes. The texture features can be applied to compute the probability of data fidelity term as a supplement. We employ the Gabor filter to extract the texture features. The image block centered at each superpixel is used to compute the output vectors of Gabor filter. The computation is implemented by the correlation coefficient with the cluster means in the texture
feature pool. The exponential form of the largest coefficient is defined as the texture probability:
\[ p(t_i|y_i) = \sup_{\mathbf{M}T_m \in T_r} \exp(r(t_i, \mathbf{M}T_m)), \quad (10) \]
where \( \mathbf{M}T_m \) denotes the \( m \)th clustering mean in the texture feature pool \( T_r \). \( r(\bullet) \) is the correlation coefficient.

The location probability is computed as follows:
\[ p(h_i|y_i) = \exp((\frac{NM_{y_i,n} + \alpha_x}{NM_{y_i} + \alpha_x})^\omega_i), \quad (11) \]
For the computation of location probability, the input image is projected onto a rectangle. \( h_i \) denotes the \( i \)th location in the rectangle. For the training set, the same projection method is applied. \( NM_{y_i,n} \) denotes the number of superpixels belonging to the label \( y_i \), which corresponds to the location \( h_i \) in the regular rectangle. \( NM_{y_i} \) denotes the total superpixel numbers in the location \( h_i \) of the regular rectangle. \( \alpha_x \) and \( \alpha_y \) are constant values. \( \omega \) varies according to the different datasets.

After the computation of data fidelity term, the smoothness term is defined by:
\[ f_{i,j}(y_i, y_j) = (1 - \delta(y_i, y_j)) \exp(-\beta \cdot d_M(i, j)) \quad (12) \]
where \( d_M(i, j) \) denotes the distance between the superpixel feature vectors:
\[ d_M(i, j) = (x_i - x_j)^T \mathbf{M}(x_i - x_j) \quad (13) \]
where \( x_i \) and \( x_j \) are the color features of \( i \)th and \( j \)th superpixels. \( \mathbf{M} \) is a positive semi-definite matrix that parameterizes the metric. The metric is Euclidean norm when \( \mathbf{M} = \mathbf{I} \). The metric is the Mahalanobis distance when \( \mathbf{M} = \Sigma^{-1} \), where \( \Sigma^{-1} \) denotes the inverse covariance matrix. \( \delta(\bullet) \) is the Kronecker delta:
\[ \delta(y_i, y_j) = \begin{cases} 0 & \text{if } y_i \neq y_j \\ 1 & \text{if } y_i = y_j. \end{cases} \quad (14) \]
The constant coefficient \( \beta \) is defined as follows:
\[ \beta = (2 < \|x_i - x_j\|^2 >)^{-1}, \quad (15) \]
where \( \bullet > \) denotes the expectation of the superpixel pairs.

Besides the global energy function, the local energy function for the \( i \)th superpixel is defined by:
\[ E_{1i}^t = -f_i(x_i|y_i) + \sum_{j \in N_{sup} (i)} \lambda_1 f_{i,j}(y_i, y_j) \]
\[ + \sum_{j \in N_{s}\{i\}} \lambda_2 f_{i,j}(y_i, y_j). \quad (16) \]

With these definitions, we apply energy minimization to obtain the optimized superpixel labels, as shown in Algorithm 1. The initial superpixel labels are specified according to the prior knowledge, e.g., the bottom 1/3 part of the image can be initialized to be the road region. Another option is to initialize the center part of the image as road region, while the non-road region is defined by:
\[ B = \{ S(i) | \min(m, n, |W - m|, |H - n|) \leq \omega, \quad (m, n) \in S(i) \}. \quad (17) \]

\[ \textbf{Algorithm 1 Superpixel MRF for image sequence} \]

**Require:**
- Input image sequence \( \{I_1, \ldots, I_M\} \);
- Superpixel set for each frame \( \{S_1, \ldots, S_M\} \);
- Threshold of the global energy function \( \varepsilon \).

1: Specify the initial superpixel label set of first frame;
2: Compute the initial global energy \( E^1 \) based on the initial superpixel labels;
3: for \( t = 2 : M \) do
4: Load the superpixel label set of time \( t - 1 \) to initialize the superpixel label set of time \( t \);
5: Compute the initial global energy function \( E^t \) of current frame according to the initial superpixel class labels. The energy function is composed of data fidelity term and smoothness term;
6: For each superpixel \( i \), compute its local energy \( E_{1i}^t \);
7: Assign superpixel \( i \) with the other class label, and compute the new local energy \( E_{1i}^t \); If \( E_{1i}^t < E_{1i}^t \), update the label: \( y_i = E_{1i}^t \);
8: Compute the updated global energy \( \hat{E}^t \) according to the new class labels;
9: if \( E^t \) and \( E^t \) are within the threshold distance of \( \varepsilon \), the algorithm terminates; Otherwise jump to Step 6;
10: end for

**Ensure:** Superpixel label set for all the image frames \( \{Y_1, \ldots, Y_M\} \).

With the assumption that \( M \) frames exist in the image sequence, the proposed superpixel MRF algorithm is shown in Alg. 1. The label initialization for the first frame is implemented by semantic labeling method [30], as shown in Fig. 3. The semantic labeling method incorporates three image segmentation methods: GraphCut [31], MeanShift [32] and Image Pyramid method [33]. Although the semantic labeling method achieves higher accurate initialization, we only apply it to initialize the first frame, for it requires too much human interaction and is very time-consuming. For the rest of the frames, the road detection process follows a cycle of “initial energy computation-local energy comparison-global energy comparison”.

The smoothness term of the energy function incorporates the superpixel interactions in both spatial and temporal domains. We assume that the road region between adjacent frames have little change, and apply the superpixel centers in previous frame to initialize the current one. Moreover, the optical flow map is utilized to match the superpixels in adjacent frames if the road region between adjacent frames changes obviously. The optical flow map is computed according to Bruhn’s method [44], which is applied to correspond the superpixel centers of current frame to the nearest centers in adjacent frames.

Based on the initial superpixel labels, the initial global
energy function $E_1$ can be computed by Eq. (7). The value of $E_t$ is the sum of the local energy function $E_i^t$. The energy function is composed of the data fidelity term and smoothness term. The data fidelity term is defined by the combination of color, texture and location probabilities. The smoothness term is defined by the interaction of spatially adjacent superpixels (Eq. 13). After the computation of the global energy function, the local energy $E_i^t$ of superpixel $i$ is specified according to Eq. (8). We then change the label of $y_i$, and compare the new local energy $\tilde{E}_i^t$ with $E_i^t$. If the new local energy gets smaller, the current class label is displaced. After the comparison of all the local energies, the new global energy $E_t$ is updated. If $\tilde{E}_i^t$ and the previous value $E_i^t$ is within a small threshold $\varepsilon$, the algorithm terminates.

The proposed algorithm can be implemented for each single image independently. In such conditions, the smoothness term of the energy function only considers about the spatial neighborhood superpixels. However, the algorithm for image sequences has the following advantages:

- Smoothness term. The smoothness term for image sequences incorporates the superpixel neighborhoods in both spatial and temporal domains.
- Initialization of the labels for the first frame. The superpixel MRF method for image sequences applies semantic labeling to initialize the first frame more accurately.
- Initialization of the labels for the rest of the frames. In the superpixel MRF method for image sequences, the superpixel labels of previous frame is propagated to current frame for initialization.

V. SPATIO-TEMPORAL SCENE RECONSTRUCTION

After the detection of the road regions in the input image sequences, the control points on road boundaries are generated to represent the road geometry. Several steps are designed to generate the control points. Firstly, we detect the vanishing points of each input image. With the assumption that the road boundaries are parallel in world coordinate system, the intersection point of the projected lines on image plane is considered to be the vanishing point. The Hough transform is applied via Gaussian sphere to detect the vanishing points [34], which decompose the task of vanishing point detection into two steps: (1) line detection and (2) line pair detection. The image plane is projected onto a unit Gaussian sphere centered at the camera origin. The intersection points of parallel lines are corresponded to the curve intersections on the Gaussian sphere.

The detected road regions are then processed for image binarization. The road boundaries are then generated based on the road and non-road regions. With the constraint of vanishing points, the potential areas of the farthest control points are identified. The intersections of road boundaries with image edges are defined as the nearest control points. The other control points are assumed to uniformly distributed between the nearest and farthest control points. The 3D corridor-style scene models are then constructed based on the control points. We apply Manhattan world assumption [42] to specify the world coordinate system. The scene models are assumed to follow the $x,y,z$ Cartesian coordinate system. Viewers can get the interpretations of road scene by changing the alignment of the system.

Based on the detection results of road regions, the background models with the 3D Corridor structure are constructed. We set the road region to be the horizontal plane, while the rest of the background regions are assumed to stand perpendicularly to the road plane. If any foreground object exists in the input image, we apply foreground/background segmentation to construct the foreground and background models independently. The foreground models are constructed with the RGBA data structure [43]. The new viewpoint images can be generated according to the view angles and viewpoint positions, as shown in Fig. 5. In order to properly arrange the 3D Corridor models, a scene model database is constructed. Each scene model corresponds to a frame in the image sequence, and a table to represent the connections between foreground and background is shown in Fig. 5 [35]. The algorithm for road scene models construction is described in Alg. 2.

Besides the scene reconstruction for monocular images, panoramic images can also be applied to the scene construction. The image stitching algorithm we used mainly includes global alignment and image blending [36]. In the global alignment stage, we apply bundle adjustment, parallax removal...

---

**Algorithm 2 Road scene models construction**

**Require:** Superpixel label sets for the image sequence: $\{Y_1, ..., Y_M\}$.

1: for $t = 1 : M$ do
2: Load the superpixel label set of frame $t$ to specify the road and non-road regions;
3: Detect the vanishing point using the Gaussian sphere method;
4: Generate the control points of road boundaries;
5: Perform the foreground segmentation and background inpainting;
6: The background scene model is constructed based on the control points;
7: The foreground scene model is constructed with polygons of RGBA data structure;
8: end for

**Ensure:** Road scene models with the 3D Corridor structure.
Fig. 4. Generation of new viewpoint images. (a) The control of view angles. The black mesh denotes the background scene, while the green mesh denotes the foreground vehicle. (b) The new viewpoint images.

Fig. 5. Scene model database of the TSD-max dataset. (a) Urban road. (b) Rural road.

and feature-based alignment to roughly stitch the input images. Although there are many projection models, the cylindrical projection model is applied. Furthermore, de-ghosting and blending steps are employed to create the output panoramas. The panoramic scene models are constructed subsequently.

VI. EXPERIMENTS AND APPLICATIONS

The platform for our experiments is a computer with Intel i5 processor @3.33GHZ and 16.0 GB RAM. The experiments for road region detection are implemented with MATLAB R14, while the scene construction experiments are based on OpenGL Toolbox. For road detection part, we design experiments using three datasets: Bristol Dataset (256 × 256 pixel size, 500 frames) [37], Caltech Dataset (320 × 240 pixel size, 1000 frames) [38] and TSD-max Dataset (256 × 256 pixel size, 1000 frames) [39]. The experiments for the scene models construction are mainly based on the TSD-max Dataset, which is constructed by the Institute of Artificial Intelligence and Robotics, Xi’an Jiaotong University. TSD-max is the basic dataset for the “Future Challenge”, a national annual competition for unmanned vehicles in China.

A. Road Detection Experiment

Fig. 6. Different segmentation methods. (a) Input image. (b) SLIC [40]. (c) TurboPixel [41]. (d) Ncut [39].

Fig. 7. Different superpixel segmentation based on Ncut method. (a) Input image. (b) N=40. (c) N=200. (d) N=500.

The proposed road detection method can be implemented for both single images (SMRF1) and image sequences (SMRF2). To quantitatively evaluate the road detection results, we apply the metrics of precision (Pre) and recall (Rec) to compare with the ground truth road regions. Precision denotes the ratio of correct pixels over the detected road region, while recall denotes the ratio of the correct pixels over the benchmark road region. In our experiments, Precision and recall are defined as follows:

\[ Pre = \frac{|R \cap R_G|}{R}, \quad Rec = \frac{|R \cap R_G|}{R_G} \]  

(18)
where $R$ and $R_G$ denote the detected region and ground truth, respectively.

F-Measure can be computed by the combination of Pre and Rec:

$$F - Measure = \frac{(1 + \alpha) \cdot Pre \cdot Rec}{\alpha \cdot Pre + Rec}$$  \hspace{1cm} (19)$$

We set $\alpha = 0.5$ to treat precision and recall equally.

In our method, superpixel segmentation is an important preliminary. In Fig. 6, three superpixel segmentation methods are compared: SLIC [40], Turbopixel [41], and Ncut [39]. The superpixel segmentation can be implemented with different scales, as shown in Fig. 7. The input images are segmented using Ncut method with the superpixel numbers of $N = 40$, $N = 200$ and $N = 500$. For the sake of feature selection and convergence speed, we choose $N = 500$ in the experiments. After the specification of superpixel sizes, the influence of iteration numbers is evaluated. The comparisons are implemented for both SMRF1 and SMRF2, using the datasets of TSD-max, Bristol and Caltech. The experiments based on different superpixel segmentations are also compared. We compute the average F-measure for every 5 iterations, using the methods of SMRF1 and SMRF2 for each image. The average F-measure over the image sequence is computed, as depicted in Fig. 9. The comparison results demonstrate that the F-measure gets more accurate with the increase of the iteration number. However, the peak values are reached at the $20_{th}$ iteration for most of the datasets. Among the three superpixel segmentation methods, Ncut method gets the most accurate results.

Furthermore, we compare the proposed methods (SMRF1 and SMRF2) with the methods of single pixel-based MRF [24], Watershed [18], Graph label [19] and GrowCut [20]. The detection results based on these state-of-the-art methods are depicted in Fig. 8, where green and blue colors denote the detected road regions and the non-road regions, respectively. The results prove that the proposed method (SMRF2) achieves the most accurate detection results from the perspective of visually pleasant. The aforementioned methods perform worse than our methods, in the circumstances that the appearance features of vehicle and road regions are too similar.

Besides the qualitative comparison, we design quantitative evaluations using the F-measure metric. The state-of-the-art methods are implemented for comparison. We compute average F-measure over the entire image sequences for each dataset, as shown in Table II. The comparison results show that the F-measures of SMRF2 are more than 0.9 for all the
### TABLE II

**Quantitative evaluation of road detection**

| Datasets | Methods   | Precision | Recall  | F-score | Time (s) |
|----------|-----------|-----------|---------|---------|----------|
| TSD-max  | PMRF      | 0.8625    | 0.8830  | 0.8692  | 1.55     |
|          | Watershed | 0.9060    | 0.9225  | 0.9114  | 0.55     |
|          | Graph label | 0.9310    | 0.9425  | 0.9348  | 1.85     |
|          | GrowCut   | 0.9410    | 0.9425  | 0.9415  | 1.36     |
|          | SMRF1+S1  | 0.9280    | 0.9410  | 0.9323  | **0.53** |
|          | SMRF1+S2  | 0.9310    | 0.9470  | 0.9363  | 0.55     |
|          | SMRF1+S3  | 0.9350    | 0.9520  | 0.9406  | 0.54     |
|          | SMRF2+S1  | 0.9395    | 0.9470  | 0.9420  | 1.28     |
|          | SMRF2+S2  | 0.9450    | 0.9525  | 0.9475  | 1.02     |
|          | SMRF2+S3  | **0.9510**| **0.9570**| **0.9530**| **1.44**|
| Bristol  | PMRF      | 0.8450    | 0.8320  | 0.8406  | 1.92     |
|          | Watershed | 0.8375    | 0.8565  | 0.8437  | 0.72     |
|          | Graph label | 0.9250    | 0.9305  | 0.9268  | 2.20     |
|          | GrowCut   | 0.9530    | 0.9410  | 0.9490  | 1.30     |
|          | SMRF1+S1  | 0.9315    | 0.9440  | 0.9356  | 0.82     |
|          | SMRF1+S2  | 0.9375    | 0.9420  | 0.9390  | **0.78** |
|          | SMRF1+S3  | 0.9385    | 0.9480  | 0.9416  | 0.84     |
|          | SMRF2+S1  | 0.9440    | 0.9485  | 0.9455  | 1.05     |
|          | SMRF2+S2  | 0.9465    | 0.9490  | 0.9473  | 1.22     |
|          | SMRF2+S3  | **0.9535**| **0.9595**| **0.9550**| 1.20     |
| Caltech  | PMRF      | 0.8210    | 0.7945  | 0.8120  | 1.40     |
|          | Watershed | 0.6250    | 0.4010  | 0.5269  | 0.92     |
|          | Graph label | 0.9310    | 0.9425  | 0.9270  | 2.40     |
|          | GrowCut   | 0.9370    | 0.9485  | 0.9408  | 1.28     |
|          | SMRF1+S1  | 0.9255    | 0.9370  | 0.9293  | **0.67** |
|          | SMRF1+S2  | 0.9205    | 0.9315  | 0.9241  | 0.82     |
|          | SMRF1+S3  | 0.9410    | 0.9400  | 0.9407  | 0.75     |
|          | SMRF2+S1  | 0.9360    | 0.9480  | 0.9400  | 1.05     |
|          | SMRF2+S2  | 0.9330    | 0.9400  | 0.9353  | 1.25     |
|          | SMRF2+S3  | **0.9410**| **0.9470**| **0.9430**| 1.32     |

3 datasets, which outperform the methods of PMRF, Watershed, Graph label, and GrowCut. This result is consistent with the qualitative results in Fig. 8. We also evaluate the results based on different superpixel segmentations, where S1, S2 and S3 denote the segmentations of TurboPixel, SLIC and Ncut, respectively. The results prove that the largest F-measures are achieved under the superpixels of Ncut. The detection time for each frame is also evaluated, as shown in the last column of Table II. As the comparison results demonstrate, the speed of SMRF1 is approximately equal to Watershed, and about 1/3 of that in PMRF. The speed of SMRF2 is approximately equal to the method of GrowCut, and outperforms that of PMRF and Graph label. In Fig. 11, more detection results with SMRF2 are depicted, where control points and vanishing points are also illustrated.

### B. Scene construction Experiment

Based on the road detection results, we then evaluate the accuracy and effectiveness of the road scene models. The scene models based on different datasets are shown in Fig. 13. We apply confusion matrix as a metric to evaluate the accuracy of the scene model components, where LW, RW, BW, RP denote “left wall”, “right wall”, “back wall” and “road plane”, as shown in Fig. 11. To further compare the proposed scene models with other models, we apply two metrics [12]: (1) Plane correctness ratio. A plane is defined as correct if more than 75% of the plane patches are correctly detected as the semantic wall and road regions. (2) Model correctness ratio. A model is defined as correct if more than 75% of patches in the wall and road planes are in the correct relationships with their neighbors. Besides, the ratio of plane patches with texture distortions is less than 25%. The evaluation was conducted by a person not associated with the project following the above metrics. We choose 2000 images with the resolution of 1024 × 1024 for the experiment.

We compare the proposed models with the methods of Make3D [12], PopUp [13] and Scene Stage [14], as shown in Fig. 12. The comparison results prove our models achieve both best plane correctness ratio and model correctness ratio. More scene construction results are shown in Fig. 13, which are based on the datasets of TSD-max, Bristol and Caltech.

In Table III, the functionalities of the proposed models are compared with that of Google Street View (GSV) [1] and Microsoft Street Slide (MSS) [2]. The comparison results prove the novelty of our system, for it has the following functionalities: (1) it is fit for panoramas; (2) the foreground objects are modeled independently; (3) free viewpoint images can be generated.

Fig. 10. Experimental results of SMRF2. (a) TSD-max dataset. (b) Bristol dataset. (c) Caltech dataset. The green color denotes the road regions, while the blue color denotes the non-road regions. The red dots refer to the control points of road boundaries, while the yellow dots refer to the vanishing points.
The applications for traffic scene simulation can be developed based on the proposed scene models, as shown in Fig. 14. We basically propose two simulation modes: (1) Bird-view mode, and (2) touring mode. In the bird-view mode, the position, direction and speed of the virtual vehicle are controlled by the users. The additional objects can be supplemented into the traffic scenes, such as obstacles, traffic signs, tower beacons, etc. Users can observe the simulation process at the bird view. In the touring mode, the new viewpoint images can be rendered with the movement of viewpoint, as shown in Fig. 4(b). The touring mode provides the users with the feelings of the road scenes.

The panoramic scene models are also constructed for the virtual street view, as shown in Fig. 15. Users can tour into the street scenes with the commands of "move forward", "move back", "turn left" and "turn right", which are consistent with the actions of a real driver. In the mean time, the touring trajectory is displayed on the map according to the GPS data.

VII. CONCLUSION AND FUTURE WORKS

In this paper, we propose a new framework to construct the spatial-temporal scene models from road image sequences. The reconstructed scene models follow a 3D corridor structure, where the road region detection is the precondition for scene construction. We design a new superpixel MRF method for road detection, following a cycle of "global energy initialization-local energy computation-global energy comparison". The data fidelity term of the energy function is defined by the combination of color, texture and location features. For the conditions of single image and image sequences, the smoothness term of the energy function is defined by the superpixel interactions of spatial and spatial-temporal superpixel neighborhoods respectively.

Based on the detection results of road regions, the control points of road boundaries are generated to construct the scene models. The scene models follow a 3D corridor structure, where the road regions are assumed to be the floor planes. The panoramic scene models are constructed to offer users...
more choices. The applications for the simulation of unmanned vehicles are developed.

In the future work, the depth map will be utilized as a supplement for scene reconstruction. With the assistance of the depth map, scene surface construction with more details can be implemented. Moreover, we will design algorithms to detect more label types, e.g. building and pedestrians.

ACKNOWLEDGMENT

This work is supported by National Natural Science Foundation of China under No. 91520301. We thank the anonymous reviewers for their valuable comments and suggestions.

REFERENCES

[1] D. Anguelov, C. Dulon, D. Filip, et al., “Google street view: Capturing the world at street level,” Computer, vol. 42, no. 6, pp. 32-38, 2010.
[2] J. Kopf, B. Chen, R. Szeliski, et al., “Street slide: Browsing street level imagery,” ACM Transactions on Graphics, vol. 29, no. 4, pp. 102-106, 2010.
[3] L. Li, D. Wen, N. Zheng, et al., “Cognitive cars: A new frontier for ADAS research,” IEEE Transactions on Intelligent Transportation Systems, vol. 13, no. 1, pp. 395-407, 2012.
[4] J. Hao, C. Li, Z. Kim, et al., “Spatio-temporal traffic scene modeling for object motion detection,” IEEE Transactions on Intelligent Transportation Systems, vol. 14, no. 1, pp. 295-302, 2013.
[5] M. Tanimoto, “Free-viewpoint television,” Signal Processing: Image Communication, vol. 27, no. 6, pp. 555-570, 2012.
[6] C. Lipski, C. Linz, K. Berger, et al., “Virtual video camera: Image-based viewpoint navigation through space and time,” Computer Graphics Forum, vol. 3, no. 2, pp. 1-11, 2010.
[7] S. Hasagasioglu, K. Kilicaslan, O. Atabay, et al., “Vehicle dynamics analysis of a heavy-duty commercial vehicle by using multibody simulation methods,” International Journal of Advanced Manufacturing Technology, vol. 60, no. 5, pp. 825-839, 2012.
[8] Y. Ganin, E. Ustinova, H. Ajakan, et al., “Domain adversarial training of neural networks,” Journal of Machine Learning Research, vol. 17, no. 59, pp. 1-35, 2016.
[9] D. Marr, S. Ullman, T. poggio, Vision: A computational investigation into the human representation and processing of visual information. MIT Press, 2010.
[10] J. Peng, J. Shen, A. Yao, et al., “Superpixel optimization using higher order energy,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 26, no. 5, pp. 917-927, 2016.
[11] Y. Horry, K. Aniyo, and K. Arai, “Tour into the picture: Using spidery interface to make animation from a single image,” ACM SIGGRAPH, pp. 225-232, 1997.
[12] A. Saxena, M. Sun, and A.Y. Ng, “Make3D: Learning 3D scene structure from a single still image,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 3, pp. 577-584, 2009.
[13] D. Hoiem, A.A. Efros, and M. Hebert, “Automatic photo pop-up,” ACM Transactions on Graphics, vol. 24, no. 3, pp. 577-584, 2005.
[14] Z. Lou, T. Gevers, N. Hu, “Extracting 3D layout from a single image using global image structures,” IEEE Transactions on Image Processing, vol. 24, no. 10, pp. 3098-3108, 2015.
[15] H. Jiang, G. Zhang, H. Wang, H. Bao, “Spatio-temporal video segmentation of static scenes and its applications,” IEEE Transactions on Multimedia, vol. 17, no. 1, pp. 3-15, 2015.
[16] ZC. Song, SG. Liu, “Sufficient image color transfer combining color and texture,” IEEE Transactions on Multimedia, vol. no., 2016.
[17] J. Shen, J. Peng, X. Dong, et al., “Higher-order energies for image segmentation,” IEEE Transactions on Image Processing, vol. 26, no. 10, pp. 4911-4922, 2017.
[18] I.J. Belaid, W. Mourou, “Image segmentation: A watershed transformation algorithm,” Image Analysis and Stereology, vol. 28, no. 1, pp. 93-102, 2009.
[19] S. Gould, J. Zhao, X. He, “Superpixel graph label transfer with learned distance metric,” European Conference on Computer Vision, pp. 632-647, 2014.
[20] K. Lu, J. Li, X. An, et al., “Vision sensor-based road detection for field robot navigation,” Sensors, vol. 15, no. 1, pp. 29594-29617, 2015.
[21] Z.L. Stan, *Markov random field modeling in image analysis*, Springer-Verlag London, 2009.

[22] I.Y. Kim, H.S. Yang, “An integration scheme for image segmentation and labeling based on Markov random field model,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 1, pp. 69-73, 1996.

[23] Y. Yang, T. Jiang, “Pixon-based image segmentation with Markov random fields,” *IEEE Transactions on Image Processing*, vol. 12, no. 12, pp. 1552-1559, 2003.

[24] C.D. Elia, G. Poggi, and G. Scarpa, “A tree-structured Markov random field model for Bayesian image segmentation,” *IEEE Transactions on Image Processing*, vol. 12, no. 10, pp. 1259-1273, 2003.

[25] A. Schick, M. Baumel, and R. Stiefelhagen, “Improving foreground segmentations with probabilistic superpixel Markov random fields,” *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 27-31, 2012.

[26] S.C. Pei, W.W. Chang, and C.T. Shen, “Saliency detection using superpixel belief propagation,” *IEEE International Conference on Image Processing*, pp. 1135-1139, 2014.

[27] X.F. Wang, X.P. Zhang, “A new localized superpixel Markov random field for image segmentation,” *IEEE International Conference on Multimedia and Expo*, pp. 1-6, 2009.

[28] J. Arrospide, L. Salgado, “Log-Gabor filters for image-based vehicle verification,” *IEEE Transactions on Image Processing*, vol. 22, no. 6, pp. 2286-2295, 2013.

[29] Y. Boykov, O. Veksler, and R. Zabih, “Fast approximate energy minimization via graph cuts,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 11, pp. 1222-1239, 2001.

[30] G.J. Brostow, J. Fauqueur, R. Cipolla, “Semantic object classes in video: A high-definition ground truth database,” *Pattern Recognition Letters*, vol. 30, no. 2, pp. 88-97, 2009.

[31] P. Felzenszwalb, D. Huttenlocher, “Efficient graph-based image segmentation,” *International Journal of Computer Vision*, vol. 59, no. 2, pp. 167-181, 2004.

[32] D. Comaniciu, P. Meer, “Mean shift: A robust approach toward feature space analysis,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603-619, 2002.

[33] P. Burt, T. Hong, and A. Rosenfield, “Segmentation and estimation of image region properties through cooperative hierarchical computation,” *IEEE Systems, Man and Cybernetics*

[34] S.T. Bernard, “Interpreting perceptive images,” *Artificial Intelligence*, vol. 21, no. 1, pp. 453-462, 1983.

[35] B.C. Russell, A. Torralba, “Building a database of 3D scenes from user annotations,” *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2711-2718, 2009.

[36] A. Eden, M. Uyttendaele, and R. Szeliski, “Seamless image stitching of scenes with large motions and exposure differences,” *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2498-2502, 2006.

[37] J. Greenhalgh, “Recognizing text-based traffic signs,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 3, pp. 1360-1369, 2015.

[38] E. Aly, “Real time detection of lane markers in urban streets,” *IEEE Intelligent Vehicles Symposium*, pp. 7-12, 2008.

[39] J.B. Shi, J. Malik, “Normalized cuts and image segmentation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888-905.

[40] R. Achanta, A. Shaji, K. Smith, et al., “SLIC superpixels compared to state-of-the-art superpixel methods,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2274-2282, 2012.

[41] S. Xiang, C. Pan, F. Nie, et al., “TurboPixel segmentation using eigenimages,” *IEEE Transactions on Image Processing*, vol. 19, no. 11, pp. 3024-3034, 2010.

[42] J.M. Coughlan, A.L. Yuille, “Manhattan world: Orientation and outlier detection by Bayesian interface,” *Neural Computing*, vol. 15, no. 5, pp. 1063-1088, 2003.

[43] Y. Li, Y. Liu, Y. Su, “Three-dimensional traffic scenes simulation from road image sequences,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 4, pp. 1121-1134, 2016.

[44] A. Bruhn, J. Weickert, C. Schn, “Lucas/Kanade meets Horn/Shunk: Combining local and global optical flow methods,” *International Journal of Computer Vision* vol. 61, no. 3, pp. 211-231, 2005.