Motion Segmentation Using Spectral Clustering on Indian Road Scenes

Mahtab Sandhu[1], Sarthak Upadhyay[2], Madhava Krishna[1], and Shanti Medasani[2]

[1] IIIT-Hyderabad, Hyderabad, India
mahtab.sandhu@research.iiit.ac.in , mkrishna@iiit.ac.in
[2] MathWorks, Hyderabad, India
{Sarthak.Upadhyay, Shanti.Medasani}@mathworks.in

Abstract. We propose a novel motion segmentation formulation over spatio-temporal depth images obtained from stereo sequences that segments multiple motion models in the scene in an unsupervised manner . The motion segmentation is obtained at frame rates that compete with the speed of the stereo depth computation. This is possible due to a decoupling framework that first delineates spatial clusters and subsequently assigns motion labels to each of these cluster with analysis of a novel motion graph model. A principled computation of the weights of the motion graph that signifies the relative shear and stretch between possible clusters lends itself to a high fidelity segmentation of the motion models in the scene.

Keywords: Motion segmentation, Object detection, Spectral clustering

1 Introduction

Motion Segmentation in cluttered and unstructured environments is challenging but pivotal for various situations that arise in autonomous driving and driver assistive systems. To do this in real-time further complicates the problem. This paper reveals a fast spatio-temporal spectral clustering formulation over stereo depths that is able to provide for both high fidelity and high rate motion segmentation on challenging native road scenes. An illustration of the output from the proposed framework can be seen in Figure 1.

The paper contributes through a robust obstacle detection algorithm, which can work in highly cluttered and unstructured environment. And a decoupled formulation, where spatial clustering is performed at dense point level to recover object level clusters that are then made temporally coherent across a subset of consecutive frames using aggregated optical tracks. Subsequently, these clusters are modeled as nodes of a motion graph where edge weights capture motion similarity among them. Finally, a spectral clustering is invoked on motion graph to recover motion models to segment moving obstacles from stationary.

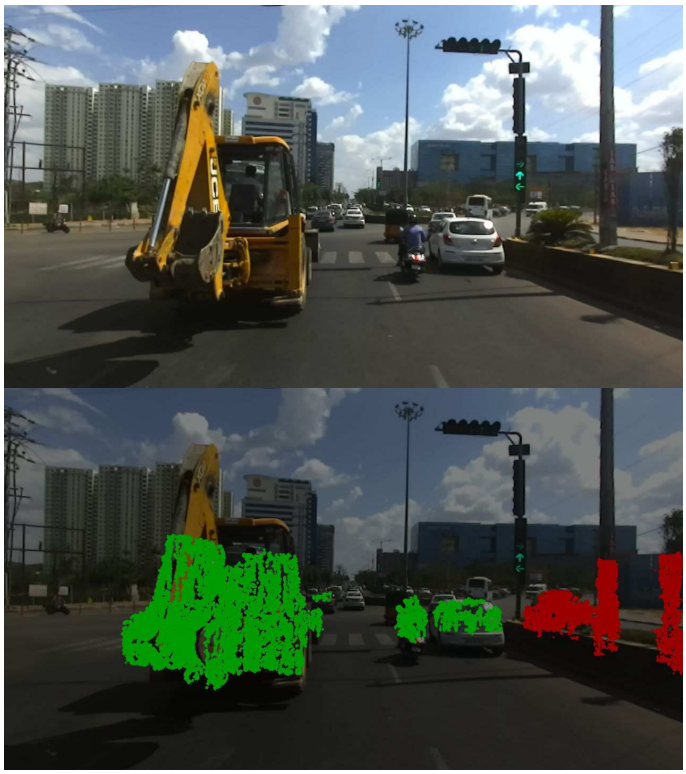

Fig. 1. (Top) Input Stereo Sequence, (Bottom) output of our motion segmentation green represents moving/dynamic and red represents static objects

It is important to note that the proposed method is independent of the label/model priors while capable of incorporating such priors when they become available. It's fidelity is not contingent on ego motion compensation or high accuracy LIDAR scan data or the availability of object and semantic priors. Semantic segmentation of these scenes itself is a challenging task due to various different types of vehicles which can be found on Indian roads. This way it contrasts itself with previous methods [1–5], a review of those is presented in the subsequent section. Comparative results vis-a-vis methods that segment motion based on stereo [1, 2] showcases the performance gain due to the present method. The paper also proposes a framework to ground-truth motion models and a metric to evaluate performance based on such a ground truth.

2 Prior Art

Sensing the environment around the host vehicle is an essential task of autonomous driving systems and advanced driver assistance systems in dynamic

environments. It forms the basis of many kinds of high-level tasks such as situation analysis, collision avoidance and path planning. Disparity based algorithm work directly on the output of stereo cameras. The V-disparity approach [6] is widely to detect road surface. This method was extended by U-V disparity method, intoduced in [7], to detect planar obstacles alongside road. Both these methods perform poorly in cluttered scenes like Fig. 3(a), 5(a) . To overcome

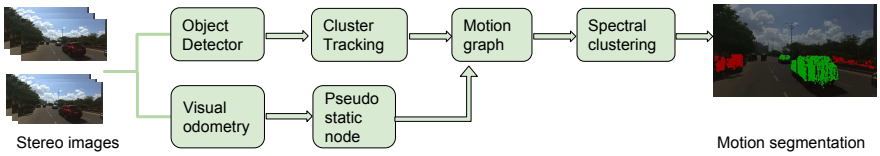

Fig. 2. The proposed pipeline for motion segmentation

this issue [6] proposed a 2-step solution to the problem of obstacle detection. In the first step they detect larger obstacles using density, subsequently they fit a road surface model on the remaining pointcloud to detect remaining obstacles. We improve this by using more robust variance feature instead of density and using a simpler road model which performs better in cluttered environment.

Once reliable obstacles are detected, classifying moving obstacles becomes imperative for path planning for autonomous driving. Among many existing ways of classifying motion segmentation methods, for the purpose of this work, we review it based on the sensing modality: monocular methods, stereo based methods and LIDAR based approaches. Existing literature has large collection of monocular motion segmentation methods [4]. Most monocular motion segmentation approaches fall in three categories: subspace clustering methods [8–12], gestalt and motion coherence based methods and optical flow cum multi view geometry based methods [13] [5] [14] [15]. The results of subspace clustering methods do not handle degeneracies such as when the camera motion follows the object typically encountered in on-road scenes wherein the motion model of the moving object lies in the same subspace as those of stationary ones. The results of such methods are typically restricted to Hopkins dataset where the degenerate scenes are not prominent. While few such as SCC [10] and [12] are able to handle the degenerate scenarios, but are limited by the prior input for number of motion models in the scene or dimensionality of motion subspaces.

Purely optical flow based methods suffer from edge effects and are erroneous in the presence of dominant flow, while the fidelity of geometry based results rely on accurate estimation of camera motion or the Fundamental matrix between scenes. Considering these limitations, recent work in [4] came up with a method based on relative shear and stretch cues as a means of combining over-segmented affine motion models into the right number of motion labels. Nonetheless, this

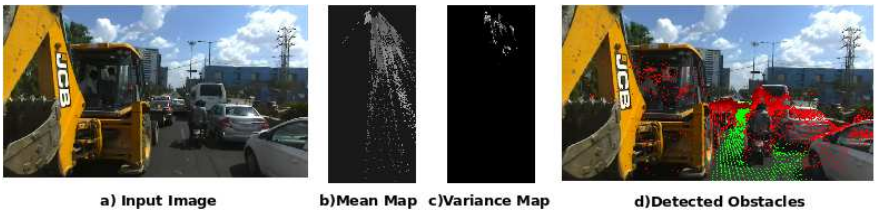

Fig. 3. Obstacle Detection in Dense Traffic: (a) Input image. (b) and (c) are the calculate mean and variance for the pointcloud in birds eye view, d) Detected Obstacles(red) and Free Space(green)

method relies on the stability of long term tracks (over 16 frames) and is not fast enough for a live outdoor application. With the advent of deep learning, recurrent neural networks (RNN) and two-stream fusion networks for joint learning of semantic and motion features have shown benchmark results for on-road driving scenes [16] [17] [18]. The deep-learning approaches however either suffer from model dependency or large running time ranging from seconds to few minutes and high computational costs involved. There are also methods based on dense LIDAR point clouds that segment motion such as in [3]. The method uses SHOT descriptors for associating point clouds, which could prove expensive for obtaining an immediate segmentation of the frames.

The closest methods to the proposed framework are [1] [2], and both use stereo depth as the primary sensing modality. While [1] segments based on clusters formed from sparse scene flow tracks, [2] uses motion potentials formed out of the divergence between predicted and obtained optical flow as the guiding principle for segmentation. The proposed method differs from both of them in terms of its philosophy by determining the number of motion models than just detecting motion regions. In terms of details, it incorporates previously segmented motion models to enhance the accuracy of the subsequent clusters, while the weights of the network are governed by the inter cluster shear and stretch cues. Since the previous methods [1], [2] detect motion but not the models of motion, we improvise our method to a motion segmentation framework and compare and contrast the advantages with respect to the prior work. While comparing with [2] we do not use the semantic cues used there but limit the comparison only based on motion cues based on flow divergence.

3 Method

We propose motion segmentation problem as spatio-temporal graph clustering similar to work done in [19] by filtering points belonging to the foreground, clustering them together spatially. We improve the foreground filtering to make it more robust in unstructured scenes. These clusters are then tracked to create a motion graph and spectral clustering is performed on this motion graph. In

order to perform motion segmentation i.e segmenting moving and static objects we add a reference node in the motion graph which mimics the motion of a static object , The cluster containing this reference node will be marked as static.

3.1 Obstacle Detection

For the task of motion segmentation we need to first segment obstacles from drivable area. The task of obstacle detection with stereo pointcloud challenging is because of the inherent noise in stereo disparity. Dense traffic of Indian road further complicates this problem. We solve this problem by a 2-step method for obstacle detection similar to [6]. We modified the algorithm to work in Indian traffic conditions.

We project the Image to the 3D space using disparity maps and divide the orthogonal 2D space relative to each frame into grids. We compute mean and variance [Fig. 3(b,c)] of 3D points belonging to each grid location and threshold it to select a grid location as belonging to foreground objects

In the next step we remove these points from the mean map [Fig. 3(c)] and fit a surface to it. The removal of high variance grid cells is critical for robust ground fitting even in dense traffic scenes where only a small portion of ground is visible.

We use a road model that allows quadratic variations of the height (Y) with the depth and linear with with the horizontal displacement. This performs better than the road model described in [19] for detection of small curbs on the side of the road.

Equation (1) shows the algebraic form of the road model, by defining the height value Y with respect to Z and X.

$$Y = aX + bZ + b'Z^2 + c \tag{1}$$

Fitting the quadratic surface to a set of n 3D points involves minimizing an error function. The error function S represents the sum of squared errors along the height:

$$S = \sum_{i=1}^{n} (Y_i - \bar{Y}_i)^2 \tag{2}$$

where Y_i is the height of the 3D point i and $\bar{Y}_i$ is the height of the surface at coordinates (X_i, Z_i). By replacing (1) into (3), the function S is obtained, where the unknowns are a, b, b, and c:

$$S = \sum_{i=1}^{n} (Y_i - (aX + bZ + b'Z^2 + c))^2 \tag{3}$$

At minimum value the partial derivatives with respect to unknowns would be zero. In matrix form

$$\begin{bmatrix} S_{X^2} & S_{XZ} & S_{XZ^2} & S_X \\ S_{XZ} & S_{Z^2} & S_{Z^3} & S_Z \\ S_{XZ^2} & S_{Z^3} & S_{Z^4} & S_{Z^2} \\ S_X & S_Z & S_{Z^2} & n \end{bmatrix} \begin{bmatrix} a \\ b \\ b' \\ c \end{bmatrix} = \begin{bmatrix} S_{XY} \\ S_{ZY} \\ S_{Z^2Y} \\ S_Y \end{bmatrix} \tag{4}$$

Equation (4) has 4 equation and four variables. We solve this using Gaussian elimination method. The estimation of road surface is done using RANSAC algorithm which gives us better fitting road surface in presence of noise. We use M points from the pointcloud to fit the surface, these are selected at random. Using more than the minimum required points to fit the surface gives us faster convergence.

3.2 Spatial Grouping

To cluster 2D points that are obtained by foreground filtering on orthographic projection of 3D points recovered from depth estimation performed over video frames [20, 21]. We adopt DBSCAN [22] for spatial clustering as it is an unsupervised density based clustering technique.Let $\mathcal{F}^1, \cdots, \mathcal{F}^\tau$ be the set of τ number of frames in a given video. For any frame $\mathcal{F}^l$ $(1 \leq l \leq \tau)$, let $\mathbf{X}^l = [\mathbf{x}_1^l, \mathbf{x}_2^l, \cdots, \mathbf{x}_n^l]$ be the set of selected n 2D points (pixels) in the image plane (i.e., $\mathbf{x} \in \mathbb{R}^2$) that belong to foreground after filtering. Let $\mathbf{Z}^l = [\mathbf{z}_1^l, \mathbf{z}_2^l, \cdots, \mathbf{z}_n^l]$ and $\mathbf{Y}^l = [\mathbf{y}_1^l, \mathbf{y}_2^l, \cdots, \mathbf{y}_n^l]$ be the respective 3D points ($\mathbf{z} \in \mathbb{R}^3$) and their 2D projection on orthographic plane ($\mathbf{y} \in \mathbb{R}^2$). We propose to incorporate prior in order to improve the performance of DBSCAN. These priors are obtained by motion model recovered in previous frame and projected to the current frame using the dense optical flow. The prior $M_{\widehat{\mathbf{y}}_i^l}$ is motion cluster to which pixel $\widehat{y}_i$ belonged in the previous frame. DBSCAN clustering by modifying the Euclidean distance metric as follows:

$$Dist(\widehat{\mathbf{y}}_i^l, \widehat{\mathbf{y}}_j^l) = \beta \; ||\widehat{\mathbf{y}}_i^l, \widehat{\mathbf{y}}_j^l|| + (1 - \beta) \; |M_{\widehat{\mathbf{y}}_i^l}, M_{\widehat{\mathbf{y}}_j^l)}| \tag{5}$$

$$if \; M_{\widehat{\mathbf{y}}_i^l} == M_{\widehat{\mathbf{y}}_i^l}, \; |M_{\widehat{\mathbf{y}}_i^l}, M_{\widehat{\mathbf{y}}_j^l)}| = 0 \; else \; |M_{\widehat{\mathbf{y}}_i^l}, M_{\widehat{\mathbf{y}}_j^l)}| = 1 \tag{6}$$

Let $\mathcal{O}^t = [\mathbf{O}_1^t, \cdots, \mathbf{O}_{c_l}^t]$ be the c_l number of clusters obtained by spatial clustering of $\widehat{\mathbf{Y}}^t$ in frame $\mathcal{F}^t$. Here, $\mathbf{O}_i^l \in \mathbb{R}^2$ is the mean vector computed over all 2D points belonging to i^{th} cluster. We interpret these clusters as individual objects present in the scene and hence call them object level clusters.

3.3 Spatio-temporal (s-t) Graph Construction

1. **Cluster Tracking:** To construct a motion graph we track the the spatially clustered objects using optical flow . 2D image Points belonging to each cluster(spatial cluster) in frame are mapped to 2d image points in the next frame using dense optical flow. We track and keep only the points whose tracks are available in more than half number of frames of the window. These common points are then projected in the 3D space and used to calculate the

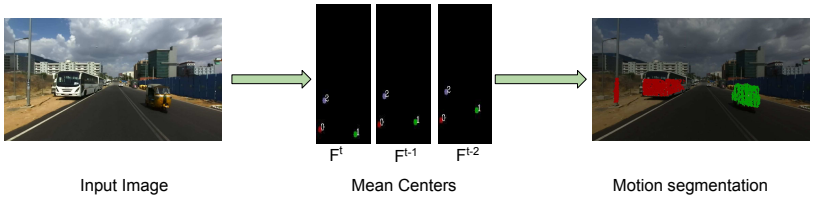

Fig. 4. Change of mean positions in orthographic space of Clusters. Note how the postion cluster 1(green) changes with respect to the other two clusters, this change is captured by the motion graph which leads to motion segmentation.

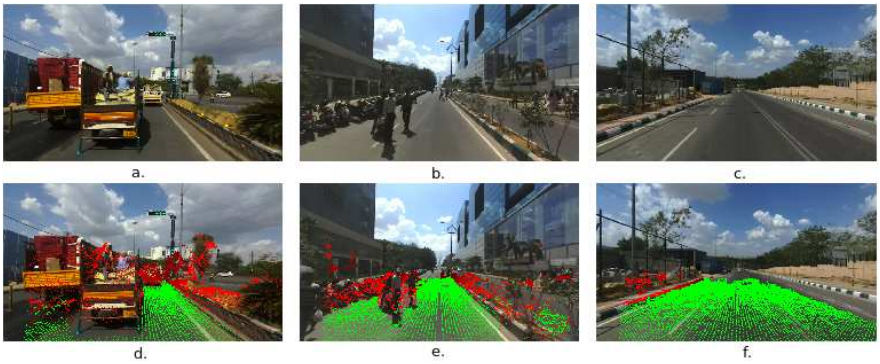

Fig. 5. Obstacle Detection on Indian roads: (a) Cluttered scene with occluded ground. (b) Pedestrians crossing road. (c) Small curbs, (d-f) Obstacle detection result, Detected Obstacles(red) and Free Space(green)

mean spatial position of the object in the 3D space . As new points may be added or subtracted from the cluster across frames which may change the motion of the mean to something different than the actual motion of the object. Using common points makes mean position of objects more stable and provides an honest motion of that cluster across frames.

2. **Creating Motion Graph:** We now construct the motion graph $\widehat{W}$ over p frames. each node in the motion graph is represented as pair of the object level clusters. Let motion graph is represented as $\widehat{\mathcal{G}}^t = \{\widehat{\mathcal{V}}^t, \widehat{\mathcal{E}}^t, \widehat{\mathcal{W}}^t\}$, and each $\widehat{\mathbf{v}}_i^t \in \widehat{\mathcal{V}}^t$ represents the motion of object center between the frame t and t-1. $\widehat{\mathbf{v}}_i^t = \{\mathbf{O}_i^t, \mathbf{O}_i^{t-1}\}$. Every pair of nodes $(\widehat{\mathbf{v}}_i^t, \widehat{\mathbf{v}}_j^t)$ will be connected by respective edge $\widehat{\mathbf{e}}_{i,j}^t \in \widehat{\mathcal{E}}^t$ with a positive valued weight $w_{i,j}^t$ capturing the motion similarity

$$w_{i,j}^t = \exp\left(-\left(\frac{d^2}{\sigma_m}\right) - \left(\frac{d_\theta^2}{\sigma_\theta}\right)\right) \tag{7}$$

$$d = \left\|\mathbf{v}_i^t - \mathbf{v}_j^t\right\| - \left\|\mathbf{v}_i^{t-1} - \mathbf{v}_j^{t-1}\right\| \tag{8}$$

$$d_\theta = \tan^{-1}\left(\mathbf{v}_i^t, \mathbf{v}_j^t\right) - \tan^{-1}\left(\mathbf{v}_i^{t-1}, \mathbf{v}_j^{t-1}\right) \tag{9}$$

Thus, for every pair of consecutive frames $\mathcal{F}^t, \mathcal{F}^{t-1}$, we would recover a motion graph $\widehat{\mathcal{G}}^t$. We propose to combine $(p-1)$ such graphs to form a single motion graph $\widehat{\mathcal{G}}$ across frames $\mathcal{F}^t, \cdots \mathcal{F}^{t-p}$ where binary edges between $\widehat{\mathbf{v}}_i^t, \widehat{\mathbf{v}}_i^{t-1}$ are assigned using the cluster level tracks.

3.4 Motion Segmentation

Spectral clustering [23] is a popular unsupervised graph clustering technique. The key idea in spectral clustering is to embed the graph by projecting each node into Euclidean space spanned by the graph Laplacian eigenvectors. Interestingly, the Euclidean distance in embedding space approximates the average connectivity on graph and therefore graph nodes that are strongly connected by paths of multiple lengths will be projected much closer and nodes that are relatively far away in connectivity space will be projected much farther. we add a pseudo static node in the motion graph. which behaves as an static object in the 3D world space. this node will act as the reference for labeling the cluster to which this node is assigned as static.

1. **Pseudo Static Node:** Visual odometry provides the rotation $\mathcal{R}$(3 x 3 matrix) and translation $\mathcal{T}$ (3 x 1 matrix) of the camera mounted on the ego vehicle between frames $\mathcal{F}^t$ and $\mathcal{F}^{t-1}$. We use libviso [24] to get $\mathcal{R}$, $\mathcal{T}$ and use it to model the motion of static objects. Any static object in camera frame(as seen in 3D reconstruction by the ego vehicle's camera) will follow inverse of ego vehicle's motion in world frame.
   We model our pseudo static node's motion between frames $\mathcal{F}^t$ and $\mathcal{F}^{t-1}$ as seen by the ego vehicle's camera as $\mathcal{X}'$ - $\mathcal{X}$. where $\mathcal{X}$ is the position of pseudo static node in frame $\mathcal{F}^{t-1}$ and $\mathcal{X}'$ is the position in frame $\mathcal{F}^t$. for the first frame $\mathcal{X}$ is initialized as $\{0,0,0,1\}^T$ in homogeneous co-ordinates and $\mathcal{X}'$ is defined as

   $$\mathcal{X}' = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix}^{-1} . \mathcal{X}$$

   similarly for each frame we calculate the position $\mathcal{X}^t$ and these tracks are then added to the motion graph. We calculate the motion of the static node independently for each window because the motion graph formulation depends on relative motion between cluster rather than actual position of the cluster.

2. **Graph Spectral clustering :** Given a weighted adjacency matrix $\widehat{\mathcal{W}}$ of a motion graph, the un-normalized graph Laplacian matrix $\mathcal{L}$ is derived as: $\mathcal{L} = \mathcal{D} - \widehat{\mathcal{W}}$, where $\mathcal{D}$ is the diagonal degree matrix of the graph with $\mathcal{D}_{i,i} = \sum_{j=1}^{n} \widehat{\mathcal{W}}_{ij}$. The K-dimensional Laplacian embedding of graph nodes

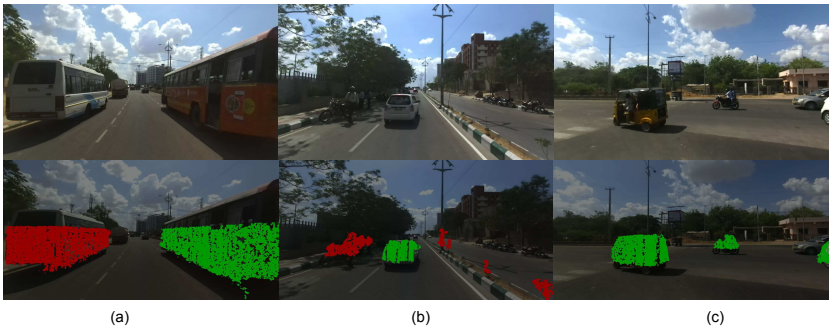

Fig. 6. Qualitative analysis of our method on Indian Sequences.

is obtained using the K eigenvectors of the graph Laplacian matrix. As stated earlier, spectral clustering involves selecting a subset of K Laplacian eigenvectors (corresponding to smallest non-zero eigenvalues) and employing K-means clustering in the embedding space to recover K clusters. The K is obtained from eigen gap analysis after getting the K clusters we mark the cluster containing the pseudo static node as static cluster and any node associated with this cluster is labeled as static. All the remaining motion models are considered as dynamic.

4 Implementation and Experiments

4.1 Experiments

We evaluate the proposed method on highly dynamic native sequences collected by us. The sequences contains dense traffic scenes and a wide variety of distinct objects/obstacles which have an even more diverse motion. Our method is able detect Fig. 5 and segment out distinct objects as moving or stationery as seen Fig. 6 (a) where a large part of image is occupied by a moving bus which disturbs the prediction of visual odometry but our method still able to segment out the two buses as static and dynamic correctly. Fig 6 (b) shows that non standard objects like bikes , pedestrians are correctly segmented as stationery and fig 6 (c) shows that the method works even without any static reference apart from the pseudo static node and segments all the moving objects as dynamic. In fig 7 (a), (b) we show that our method is able to segment the staionary cars correctly and in fig 7 (c) it is able so detect and distinguish between walking and stationary pedestrians.

4.2 Implementation

The method was implemented in C++ and tested on an intel i7 , 3.0Ghz processor. We found that using a window size of 3 gave us the best trade off between

Fig. 7. Note here major part of the image is occupied by objects which makes object detection difficult and subsequently makes motion segmentation difficult

motion segmentation accuracy and time taken per frame.Our approach takes around 180 ms per frame for motion segmentation. For the motion graph construction we used $\sigma_1 = 0.01$ and $\sigma_2 = 0.04$. The dataset was collected using a ZED Stereo Camera with baseline of 12cm at 30 fps.

Method	**Motion Accuracy(%)**
SCENE-M [1]	72.51
FLOW-M [2] + DIS [25] + SGBM [20]	61.57
FLOW-M [2] + DIS [25] + BM [21]	62.73
FLOW-M [2] + DeepFlow [26] + BM [20]	67.38
FLOW-M [2] + DeepFlow [26] + SGBM [21]	69.11
ours	81.03

Table 1. Quantitative motion segmentation evaluation of our proposed approach against SCENE-M [1] , FLOW-M [2] and SSD - M [19] with different priors

5 Conclusions

This paper proposed a method to segment motion through spectral decomposition . We make use of the motion model clustering framework described in [19] and adapt it to perform motion segmentation in unstructured Indian scenes.We propose our novel object detector which works even with dense traffic where very less ground plane is visible thus providing accurate objects for tracking

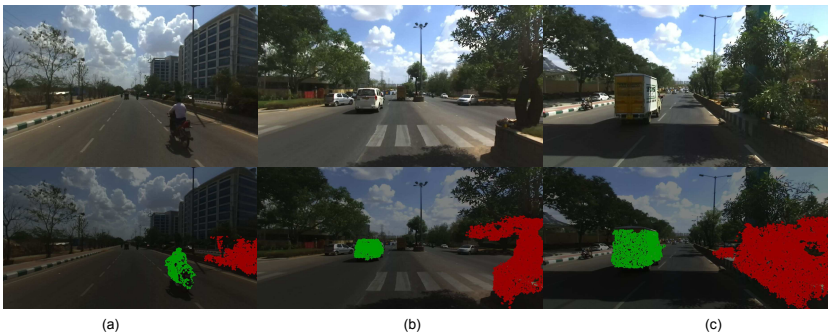

Fig. 8. Qualitative Evaluation on Indian on-road sequences

and Spectral clustering. Using visual odometry as a reference static node we are successfully able to perform motion segmentation . The method works even without accurate visual odometry because spectral clustering projects the the pseudo static node in euclidean space where it will be nearer to ground truth static motion model thus making it robust in highly dynamic and diverse conditions .

6 Acknowledgement

The work described in this paper is supported by MathWorks. The opinions and views expressed in this publication are from the authors, and not necessarily that of the funding bodies

References

1. Lenz, P., Ziegler, J., Geiger, A., Roser, M.: Sparse scene flow segmentation for moving object detection in urban environments. In: IV, IEEE (2011) 926–932
2. Reddy, N.D., Singhal, P., Chari, V., Krishna, K.M.: Dynamic body vslam with semantic constraints. In: IROS 2015
3. A.Dewan, Caselitz, T., Tipaldi, G., Burgard, W.: Motion-based detection and tracking in 3d lidar scans. In: ICRA. (2016)
4. Tourani, S., Krishna, K.M.: Using in-frame shear constraints for monocular motion segmentation of rigid bodies. JIRS (2016)
5. Namdev, R.K., Kundu, A., Krishna, K.M., Jawahar, C.: Motion segmentation of multiple objects from a freely moving monocular camera. In: ICRA. (2012) 4092–4099
6. Oniga, F., Nedevschi, S.: Processing dense stereo data using elevation maps: Road surface, traffic isle, and obstacle detection. IEEE Transactions on Vehicular Technology **59**(3) (2010) 1172–1182
7. Hu, Z., Uchimura, K.: Uv-disparity: an efficient algorithm for stereovision based scene analysis. In: Intelligent Vehicles Symposium, 2005. Proceedings. IEEE, IEEE (2005) 48–54
8. Lauer, F., Schnörr, C.: Spectral clustering of linear subspaces for motion segmentation. In: ICCV. (2009)
9. Elhamifar, E., Vidal, R.: Sparse subspace clustering. In: CVPR. (2009) 2790–2797
10. Chen, G., Lerman, G.: Spectral curvature clustering (scc). IJCV **81**(3) (2009) 317–330
11. Jain, S., Madhav Govindu, V.: Efficient higher-order clustering on the grassmann manifold. In: ICCV. (2013) 3511–3518
12. Zappella, L., Provenzi, E., Lladó, X., Salvi, J.: Adaptive motion segmentation algorithm based on the principal angles configuration. In: ACCV. (2010)
13. Kundu, A., Krishna, K., Sivaswamy, J.: Moving object detection by multi-view geometric techniques from a single camera mounted robot. In: IROS. (2009)
14. Vidal, R., Sastry, S.: Optimal segmentation of dynamic scenes from two perspective views. In: CVPR. Volume 2. (2003)
15. Ochs, P., Brox, T.: Higher order motion models and spectral clustering. In: CVPR. (2012)
16. Vertens, J., Valada, A., Burgard, W.: Smsnet: Semantic motion segmentation using deep convolutional neural networks. In: IROS. (2017)
17. Haque, N., Reddy, D., Krishna, M.: Joint semantic and motion segmentation for dynamic scenes using deep convolutional networks. In VISAPP (2017)
18. Lin, T.H., Wang, C.C.: Deep learning of spatio-temporal features with geometric-based moving point detection for motion segmentation. In: ICRA. (2014) 3058–3065
19. Sandhu, M., Haque, N., Sharma, A., Krishna, K.M., Medasani, S.: Fast multi model motion segmentation on road scenes. In: Intelligent Vehicles Symposium (IV), 2018 IEEE, IEEE (2018) 2131–2136
20. Furht, B., ed. In: Block Matching. Springer US, Boston, MA (2008) 55–56
21. Hirschmuller, H.: Stereo processing by semiglobal matching and mutual information. (2008)
22. Ester, M., Kriegel, H.P., Sander, J., Xu, X.: A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise. In: KDDM. (1996)

23. Luxburg, U.: A tutorial on spectral clustering. Statistics and Computing **17**(4) (2007) 395–416
24. Geiger, A., Ziegler, J., Stiller, C.: Stereoscan: Dense 3d reconstruction in real-time. In: Intelligent Vehicles Symposium (IV). (2011)
25. Kroeger, T., Timofte, R., Dai, D., Van Gool, L.: Fast optical flow using dense inverse search. In: ECCV. (2016)
26. Weinzaepfel, P., Revaud, J., Harchaoui, Z., Schmid, C.: Deepflow: Large displacement optical flow with deep matching. In: ICCV. (2013)