Fast Multi Model Motion Segmentation on Road Scenes

Mahtab Sandhu, Nazrul Haque, Avinash Sharma, K Madhava Krishna and Shanti Medasani

***Abstract*— We propose a novel motion clustering formulation over spatio-temporal depth images obtained from stereo sequences that segments multiple motion models in the scene in an unsupervised manner. The motion models are 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. The fidelity is vindicated through accuracies reaching 89.61% on KITTI and complex native sequences.**

I. INTRODUCTION

Segmenting a video sequence into multiple motion models is pivotal for various situations that arise in autonomous driving and driver assistive systems. To obtain such motion models at high frame rates has typically proven to be challenging and nearly elusive. This paper reveals a new spatio-temporal spectral clustering formulation over stereo depths that is able to provide for both high fidelity and high frame rates motion model segmentation on KITTI [1] and challenging native road scenes. An illustration of the output from the proposed framework can be seen in Figure 1

The paper contributes through the novel 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. 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. This way it contrasts itself with previous methods [2], [3], [4], [5], [6], a review of those is presented in the subsequent section. Comparative results vis-a-vis methods that segment motion based on stereo [2], [3] 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.

All authors are associated with Robotics Research Center, International Institute of Information Technology, Hyderabad, India mahtab.sandhu@research.iiit.ac.in

Fig. 1. (Top) Input RGB Stereo Sequence, (Bottom) Eventual output of our multi-motion model segmentation approach. Each color represents individual object motion and objects with the same motion model are represented with the same color such as the two bikes to the left of the image (seen with *violet* color).

II. PRIOR ART

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 [5]. Most monocular motion segmentation approaches fall in three categories: subspace clustering methods [7], [8], [9], [10], [11], gestalt and motion coherence based methods and optical flow cum multi view geometry based methods [12] [6] [13] [14]. 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 [9] and [11] 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 [5] came up with a method based on relative shear and stretch cues as a

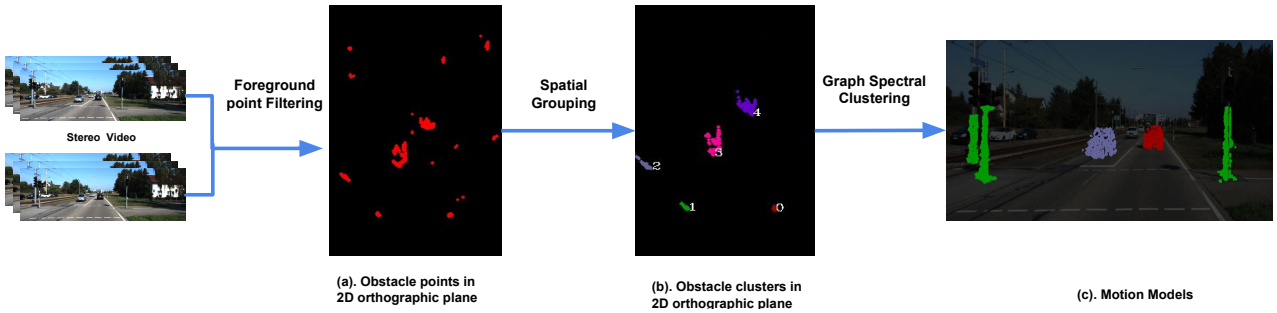

Fig. 2. Outline of the proposed motion segmentation method.

Fig. 3. Mapping between image points ($\mathbf{X}^i$) and there corresponding points in 2D orthographic plane($\mathbf{Y}^i$)

means of combining over-segmented affine motion models into the right number of motion labels. Nonetheless, this 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 [15] [16] [17]. 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 [4]. 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 [2] [3], and both use stereo depth as the primary sensing modality. While [2] segments based on clusters formed from sparse scene flow tracks, [3] 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 [3] [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 [3] we do not use the semantic cues used there but limit the comparison only based on motion cues based on flow divergence. Specifically, we show a performance gain in terms of accuracy to the tune of 11.6% vis-a-vis the prior art.

III. METHOD

We propose motion model segmentation problem as spatio-temporal graph clustering, thereby capturing the relative motion of different objects over a sequence of frames. The proposed solution first performs a foreground point filtering followed by the spatial clustering of points to recover object level clusters and later the motion segmentation is eventually obtained with spectral clustering employed over object level motion graphs.

A. Foreground Point Filtering

Only a subset of 2D points in a given scene belong to foreground (moving objects). In order to make our method work in generic scenarios and use less computing resources, we project the Image to the 3D space and divide the orthogonal 2D space relative to each frame into grids.Objects belonging to the foreground will have higher mean height. We compute basic mean and variance of 3D points belonging to each grid location and threshold it to select a grid location as belonging to foreground objects. We merge multiple image points to single grid point. Nevertheless, we always have a one-to-one mapping from 2D image points to 3D points and many-to-one mapping from 3D points to grid points in 2D orthographic plane as seen in figure 3. Thus, we can easily transfer the dense optical tracks obtained from image pixels to grid level tracking.

B. 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 [18], [19]. We adopt DBSCAN [20] 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

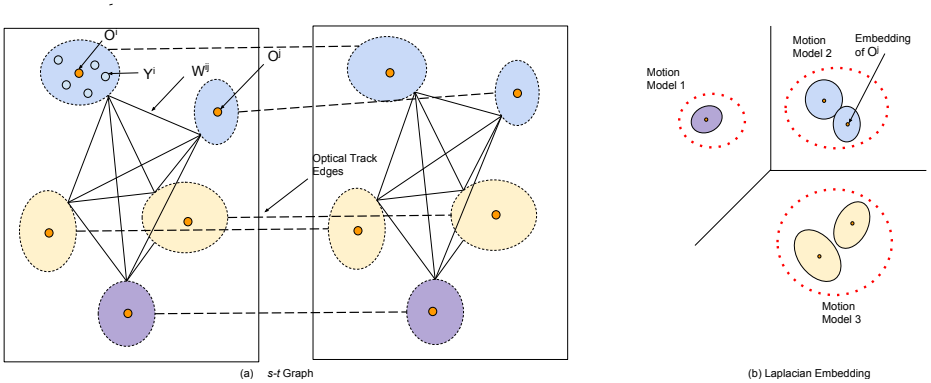

Fig. 4. (a) *s-t* graph model computed over three consecutive frames. (b) Laplacian embedding of graph nodes.

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:

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

$$if \; M_{\widehat{\mathbf{y}}_i^l} == M_{\widehat{\mathbf{y}}_i^l}, \; |M_{\widehat{\mathbf{y}}_i^l}, M_{\widehat{\mathbf{y}}_j^l)}| = 0 \; else \; |M_{\widehat{\mathbf{y}}_i^l}, M_{\widehat{\mathbf{y}}_j^l)}| = 1 \quad (2)$$

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.

C. Spatio-temporal (s-t) Graph Construction

(a) **Cluster Tracking:** In order to relate the independent clusters obtained in each frame, we propose a cluster based tracking using optical flow. Dense optical flow is pre-computed using Dense Inverse Search algorithm[21]. The spatial cluster $\mathbf{O}_i^t$ with maximum number of optically tracked points is taken as the tracked cluster for $\mathbf{O}_i^{t-1}$.This object level cluster tracking provides us reliable tracks as compared to pixel level tracking. If no such cluster is found or the matched points are below a certain threshold, either due to temporary occlusion or disappearance, we handle the case by introducing pseudo clusters. We estimate and assign the orthogonal coordinates of the pseudo spatial cluster using a running average of the 2D positions of the tracked clusters in the previous frames. The stated approach is repeated with reverse iteration, i.e, from frame $\mathcal{F}^t$ to $\mathcal{F}^{t-p}$, with backward cluster tracking. Additionally, we remove spatial clusters which do not appear in significant number of frames specifically, $\lfloor p/2 \rfloor$ in the given frame window of p frames, thus qualifying as a case of late appearance or early disappearance.

(b) **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

$$w_{i,j}^t = \exp\left(-\left(\frac{d^2}{\sigma_m}\right) - \left(\frac{d_\theta^2}{\sigma_\theta}\right)\right) \quad (3)$$

$$d = \left\|\mathbf{v}_i^t - \mathbf{v}_j^t\right\| - \left\|\mathbf{v}_i^{t-1} - \mathbf{v}_j^{t-1}\right\| \quad (4)$$

$$d_\theta = \tan^{-1}\left(\mathbf{v}_i^t, \mathbf{v}_j^t\right) - \tan^{-1}\left(\mathbf{v}_i^{t-1}, \mathbf{v}_j^{t-1}\right) \quad (5)$$

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.Figure 4(a) depicts the construction of spatio-temporal graph stitched across multiple frames.

D. Graph Spectral Clustering

The key idea in spectral clustering [22] is to embed the graph by projecting each nodes into Euclidean space spanned by the graph Laplacian eigenvectors. the Euclidean distance in embedding space approximates connectivity on the graph.Spectral clustering involves selecting a subset of K Laplacian eigenvectors (corresponding to smallest non-zero eigenvalues) and employing Kmeans clustering in the embedding space to recover K clusters. The number of clusters is typically a user given parameter in case of Kmeans algorithm. However, in case of spectral clustering, the eigengap detection [22] can be used to automatically recover the number of clusters (K). The idea of eigengap analysis is to find the sharpest increase in eigenvalue (i.e., $max(\lambda_{i+1} - \lambda_i)$) and selecting the corresponding index as the the value of K. 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 is obtained

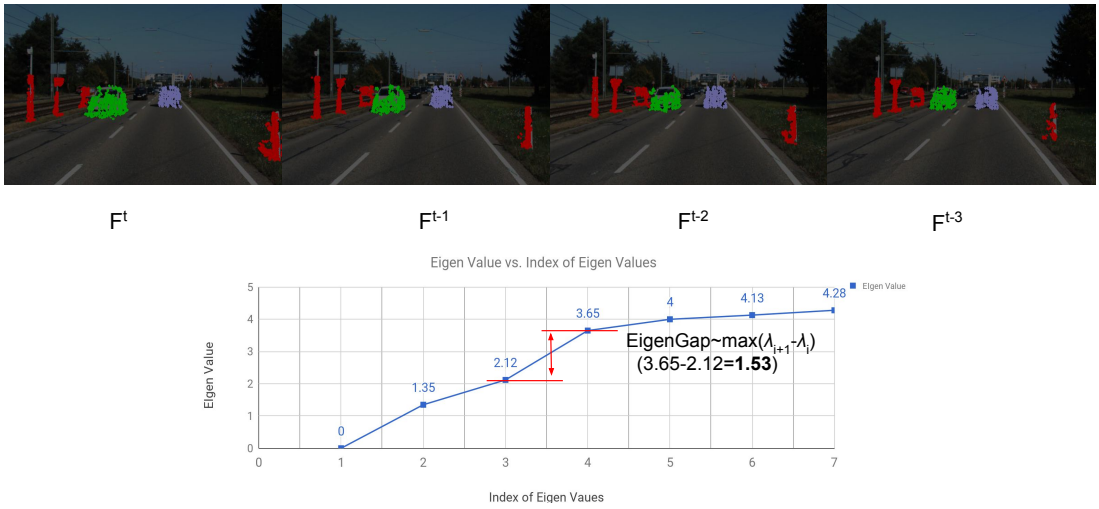

Fig. 5. (Bottom) Eigengap analysis; (Top) Corresponding motion segmentation results with three motion models.

using the K eigenvectors of the graph Laplacian matrix. Let, $\mathcal{L} = \mathbf{U}\Lambda\mathbf{U}^T$ be the eigen-decomposition of the $\mathcal{L}$ matrix where $\mathbf{U} = [\mathbf{u}_1, \cdots, \mathbf{u}_n]$ be the eigenvectors and $\Lambda = Diag(\lambda_1, \cdots, \lambda_n)$ be the corresponding eigenvalues of the L matrix with property that $\{\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n\}$ and $\lambda_1 = 0, \mathbf{u}_1 = \vec{1}$ for a connected graph. Kmeans clustering is then employed in the embedding space to recover the K motion models. Figure 5 show a real eigenvalue plot where eigengap detection yields $K = 3$ (excluding zero eigenvalue/vector pair) and corresponding motion segmentation results over the sequence of four frames where three motion models are shown with different colors. Thus, the proposed algorithm automatically finds the number of motion models and provide motion model labels for each image pixel.

IV. EXPERIMENTS & RESULTS

For the evaluation of our approach we use the renown KITTI-Tracking benchmark [1]. In order to evaluate the accuracy of our method in terms of number of motions (motion models) in a scene, we propose a means of ground truthing motion models. While KITTI dataset provides a variety of benchmarks it does not provide one that indicates the number of motions in a scene.

A. Ground Truthing Motion Model

We use the 3D bounding boxes provided by the tracking benchmark to generate the Ground Truth motion model for each object in the scene. We iterate over a window of five frames and calculate the object motion using the tracklet coordinates for each pair of frame encountered. We capture the motion model of the object by computing its average motion over the window. The eventual motion model of the sequence is obtained as number of clusters, where each cluster contains objects with similar individual motions. To showcase proficiency and robustness of our approach in diverse settings, we also evaluate the proposed approach on Indian on-road scenes. The native sequences proves challenging with regard to both variability in type of objects encountered in the scene and adverse scene conditions observed. The Indian road dataset consists of scenes with uncommon objects such as bikes, three-wheelers, trucks, and other atypical moving vehicles. We manually annotate 120 frames on the Indian dataset For all experiments on native sequences, we use color RGB-images with a resolution of 1280 x 600.

Method	Tight Accuracy Model(%)				Relaxed Accuracy Model(%)			
	Seq - 3	Seq - 5	Seq - 10	Seq - 11	Seq - 3	Seq - 5	Seq - 10	Seq - 11
SSD	83.66	92.05	93.81	76.84	82.31	88.76	95.78	85.88
SSD-M	84.73	92.54	94.47	78.02	87.56	94.67	96.66	86.29
SDD	86.43	89.17	95.78	77.23	90.02	92.61	97.57	86.41
SDD-M	88.13	91.11	96.66	82.59	92.85	95.14	98.25	89.24

TABLE I

QUANTITATIVE MOTION MODEL EVALUATION ON KITTI TRACKING SEQUENCES WITH RESPECT TO GROUND TRUTH-ED MOTION MODELS.

B. Motion Model Evaluation

In this section, we analyze the performance of our motion model segmentation against the Ground-Truth motion model. We cover our evaluation using two notable disparity computation algorithms Block Matching (BM)[18] and Semi-Global Block Matching (SGBM)[19] abbreviated as SSD and SDD respectively. Similarly, SSD-M denotes Spectral Clustering over Sparse Disparity with Motion Prior (as described in Equation 1) and SDD-M, Spectral Clustering over Dense Disparity with Motion Prior. We use four prominent KITTI-Tracking sequences. Table I and Table II summarizes our multi motion model evaluation. Specifically, Sequence 3, 5 and 10 represent highway sequences, while Sequence 11 was chosen to evaluate the proposed method on inner-city traffic scenes with multiple moving agents. We split our evaluation with two ways of computing the accuracy vis-a-vis Ground Truth.

1) Tight Accuracy Model: Here, we denote per-frame motion model accuracy as the ratio of number of motion models classified correctly to the total number of motion models present in the scene. A motion model is said to be correctly

(a) (b) (c)

Fig. 6. Qualitative Evaluation on KITTI dataset. The results show proficiency of our approach across diverse scenes with challenging conditions. Please note that stationary objects above a certain height such as poles get grouped as a motion model – the stationary model

(a) (b) (c)

Fig. 7. Qualitative Evaluation on Indian on-road dataset.

Method	Tight Accuracy(%)	Relaxed Accuracy(%)
SSD (ours)	80.38	90.82
SSD-M (ours)	82.71	90.33
SDD (ours)	84.47	90.48
SDD-M (ours)	86.51	91.26

TABLE II
QUANTITATIVE MOTION MODEL EVALUATION ON INDIAN ON-ROAD SEQUENCES

Method	Motion Accuracy(%)
SCENE-M [2]	72.51
FLOW-M [3] + DIS [21] + SGBM [18]	61.57
FLOW-M [3] + DIS [21] + BM [19]	62.73
FLOW-M [3] + DeepFlow [23] + BM [18]	67.38
FLOW-M [3] + DeepFlow [23] + SGBM [19]	69.11
SSD - MS (ours)	80.03
SDD - MS (ours)	82.65

TABLE III
QUANTITATIVE MOTION SEGMENTATION EVALUATION OF OUR PROPOSED APPROACH AGAINST SCENE-M [2] AND FLOW-M [3]

classified if and only if all the objects following the motion model(as per the Ground Truth) are clustered together. We are able to obtain significant increase in accuracy when using the motion priors from previous frames. The increase in more prominent in Sequence 11 owing to the multiple moving objects present in the scene In general, the results are more accurate in SDD-M than SSD-M due to better disparity maps obtained from the algorithm. The high accuracies obtained in diverse scenes are attributed to the inter cluster shear and stretch cues incorporated with the motion model priors in a unified spectral framework. This further shows the adaptability of our approach in diverse settings by bypassing the need of an object detector in challenging scenarios.

2) Relaxed Accuracy Model : The evaluation metric is a relaxed version of *Motion Model* based analysis explained in the previous section. Instead of assigning a binary score to correctness of a particular motion model, we compute the ratio of maximum number of objects clubbed together to the total number of objects assigned to the motion model (as per the GT). The remaining formulation remains the same as that of *Motion Model* based evaluation.

C. Motion Segmentation Accuracy

The proposed method can only recover relative motion models but to compare our method with other methods that recover only dynamic and static motion models. We add pseudo object node that represents static objects which can be initialized and updated using the ego-motion estimation. We use the KITTI Ground Truth motion dataset provided by [16] with 100 annotated images from KITTI Tracking benchmark. To the best of our knowledge, the prominent methods showcasing results for motion segmentation on stereo sequences are SCENE-M [2] and FLOW-M [3]. We quantitatively compare our method with [2] and [3] on the annotated dataset. We compute the moving object detection accuracy as the ratio of the moving objects detected to the total number of moving objects in the scene. An object is classified as moving if the majority number of points lying on the object are predicted as moving by the algorithm. We

use the bounding boxes from the KITTI-Tracking Ground Truth for representing an object in the scene. Table III shows the motion segmentation accuracy obtained from our spectral clustering approach over dense (SDD-MS) and sparse disparity(SSD-MS) in comparison to SCENE-M and FLOW-M which we implemented ourself. We are able to obtain an increase of 10.14% in accuracy over SCENE-M and outperform FLOW-M with both DeepFlow [23] and Fast-DIS[21] as input flow to the algorithm and attain an overall accuracy of 82.64% in detecting moving objects on-road. Our approach clearly outperforms the existing approaches by a significant margin due to the shear cues incorporated in an unsupervised clustering approach for segmenting moving objects.

Method	Time (ms)
SDD -MS	589
SDD	387
SSD-MS	186
SSD	107

TABLE IV

PREDICTION TIME ANALYSIS OF OUR APPROACH

D. Prediction Time Analysis

Real time and fast prediction is one of major attribute needed when looking at real-world implementation of motion prediction based approaches. Especially in the context of Autonomous Driving and Driver Assistance Systems detecting motion models needs to be achieved almost at capture frequency. We evaluate our running time for both multi motion model and motion segmentation on a single core CPU of an intel i7 processor at @ 2.50GHz and the time taken is reported in Table IV. Our approach takes 107 ms for multi-motion model segmentation and 186 ms for segmenting moving objects in the scene, thereby enabling low cost and efficient implementation with competitive accuracy. To the best of our knowledge this is the first such reporting of the speed at which motion models and motion segmentation is computed amongst the methods surveyed by the authors.

E. Discussion

In the proposed method object clusters are tracked over a few frames and modeled as the nodes of motion graph. The length of the tracking window is kept short as the objects can be accelerating, which changes their motion very fast hence a smaller window provides more accurate segmentation. As no object detector are used edges of detected objects are not sharp but easily distinguishable. Spectral clustering is invoked on motion graph to recover motion models.Note that two motion models will get clubbed together if difference in their motion is less than the minimum resolution of the motion graph as the eigen gap would not be enough to distinguish between them.

V. CONCLUSIONS

This paper proposed a novel method to recover motion models through spectral decomposition methods at frame rates nearly equal to that of disparity computation. The present formulation follows a decoupled approach, wherein spatial clusters are formed (aided with motion priors) and a motion graph over such clusters is applied with spectral clustering to find the number of motions in the scene. The paper reports accuracies that exceed more than 90% on some of the KITTI sequences as well as on complex Indian road scenes, which contain apart from cars a variety of two wheeled and three wheeled moving objects.

VI. ACKNOWLEDGEMENT

The work described in this paper is supported by MathWorks, Hyderabad

REFERENCES

[1] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *CVPR*, 2012.

[2] P. Lenz, J. Ziegler, A. Geiger, and M. Roser, "Sparse scene flow segmentation for moving object detection in urban environments," in *IV*. IEEE, 2011, pp. 926–932.

[3] N. D. Reddy, P. Singhal, V. Chari, and K. M. Krishna, "Dynamic body vslam with semantic constraints," in *IROS 2015*.

[4] A.Dewan, T. Caselitz, G. Tipaldi, and W. Burgard, "Motion-based detection and tracking in 3d lidar scans," in *ICRA*, 2016.

[5] S. Tourani and K. M. Krishna, "Using in-frame shear constraints for monocular motion segmentation of rigid bodies," *JIRS*, 2016.

[6] R. K. Namdev, A. Kundu, K. M. Krishna, and C. Jawahar, "Motion segmentation of multiple objects from a freely moving monocular camera," in *ICRA*, 2012, pp. 4092–4099.

[7] F. Lauer and C. Schnörr, "Spectral clustering of linear subspaces for motion segmentation," in *ICCV*, 2009.

[8] E. Elhamifar and R. Vidal, "Sparse subspace clustering," in *CVPR*, 2009, pp. 2790–2797.

[9] G. Chen and G. Lerman, "Spectral curvature clustering (scc)," *IJCV*, vol. 81, no. 3, pp. 317–330, 2009.

[10] S. Jain and V. Madhav Govindu, "Efficient higher-order clustering on the grassmann manifold," in *ICCV*, 2013, pp. 3511–3518.

[11] L. Zappella, E. Provenzi, X. Lladó, and J. Salvi, "Adaptive motion segmentation algorithm based on the principal angles configuration," in *ACCV*, 2010.

[12] A. Kundu, K. Krishna, and J. Sivaswamy, "Moving object detection by multi-view geometric techniques from a single camera mounted robot," in *IROS*, 2009.

[13] R. Vidal and S. Sastry, "Optimal segmentation of dynamic scenes from two perspective views," in *CVPR*, vol. 2, 2003.

[14] P. Ochs and T. Brox, "Higher order motion models and spectral clustering," in *CVPR*, 2012.

[15] J. Vertens, A. Valada, and W. Burgard, "Smsnet: Semantic motion segmentation using deep convolutional neural networks," in *IROS*, 2017.

[16] N. Haque, D. Reddy, and M. Krishna, "Joint semantic and motion segmentation for dynamic scenes using deep convolutional networks," *In VISAPP*, 2017.

[17] T.-H. Lin and C.-C. Wang, "Deep learning of spatio-temporal features with geometric-based moving point detection for motion segmentation," in *ICRA*, 2014, pp. 3058–3065.

[18] B. Furht, Ed., *Block Matching*. Boston, MA: Springer US, 2008, pp. 55–56.

[19] H. Hirschmuller, "Stereo processing by semiglobal matching and mutual information," 2008.

[20] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise," in *KDDM*, 1996.

[21] T. Kroeger, R. Timofte, D. Dai, and L. Van Gool, "Fast optical flow using dense inverse search," in *ECCV*, 2016.

[22] U. Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, 2007.

[23] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid, "Deepflow: Large displacement optical flow with deep matching," in *ICCV*, 2013.