Unsupervised Learning of Shape-Motion Patterns for Objects in Urban Street Scenes

Dirk Klostermann
dirk.klostermann@rwth-aachen.de
Aljoša Ošep
osep@vision.rwth-aachen.de
Jörg Stückler
stueckler@vision.rwth-aachen.de
Bastian Leibe
leibe@vision.rwth-aachen.de

Computer Vision Group,
Visual Computing Institute,
RWTH Aachen University
Aachen, Germany

Abstract

Tracking in urban street scenes is predominantly based on pretrained object-specific detectors and Kalman filter based tracking. More recently, methods have been proposed that track objects by modelling their shape, as well as ones that predict the motion of objects using learned trajectory models. In this paper, we combine these ideas and propose shape-motion patterns (SMPs) that incorporate shape as well as motion to model a variety of objects in an unsupervised way. By using shape, our method can learn trajectory models that distinguish object categories with distinct behaviour. We develop methods to classify objects into SMPs and to predict future motion. In experiments, we analyze our learned categorization and demonstrate superior performance of our motion predictions compared to a Kalman filter and a learned pure trajectory model. We also demonstrate how SMPs can indicate potentially harmful situations in traffic scenarios.

1 Introduction

Analyzing and predicting the movement of objects is a vital ability for self-driving cars or autonomous mobile robots. Such systems need to be able to foresee potential collisions and also react to possibly harmful situations. A common approach in many state-of-the-art vision systems is to base motion prediction on tracking and to formulate object tracking as inference in linear dynamical systems such as Kalman filters. Since these trackers typically assume simple physical motion models, they cannot take into account the versatile set of motion behaviors of the various object categories in an environment. For instance, these models do not include potential changes in behavior such as objects that suddenly start moving.

In urban street scenes, the variety of possible objects and their motion patterns limits the feasibility of manually engineered parametric models of motion behaviour. We propose an unsupervised approach to learn motion patterns of object categories from example data. In our approach, object categories are not limited to predefined classes. Instead, our approach provides a categorization into objects with similar shapes and trajectories.

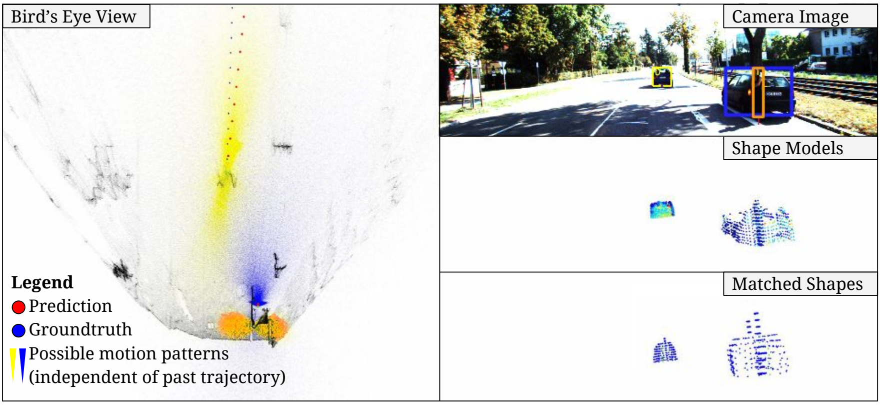


Figure 1: Possible future motion predictions based on shape-motion patterns (SMPs). In this scene, a car (yellow) is driving forward, another car (blue) is parked on the right and next to the parked car a pedestrian (orange) is waiting. Our model predicts possible motion behaviour even for static objects. Note that in this situation our model can indicate that the standing pedestrian may walk onto the road.

For the shape of an object, our approach predicts a distribution over possible future motions. Based on this information, autonomous cars could assess, for example, if other traffic participants could potentially interfere with the autonomous car. Fig. 1 illustrates an example, in which our model provides informative predictions on the possible future motions of cars and pedestrians in a traffic scenario.

Our approach is based on object tracks which are represented by 3D shape and position trajectories. We propose to cluster these tracks hierarchically: first into clusters of similar shapes, then into typical motion patterns within a shape cluster. These learned shape-motion patterns (SMPs) allow for predicting the future motion of objects and for modelling their possible future behaviours based on their shape. In experiments, we evaluate the accuracy of our motion prediction and demonstrate superior performance towards a Kalman filter based tracker and a learned pure trajectory model. We also analyse the categorization found by our unsupervised learning approach.

In summary, we make the following contributions in our work: (1) We propose a method for unsupervised learning of shape-motion patterns of objects in urban street scenes from stereo video. (2) We use the learned patterns to predict the future motion of novel objects from the learned categories. (3) Our approach is not restricted to specific shapes or moving objects. From a shape, it can model possible future motions – even for static objects.

2 Related work

Current research on motion prediction can be structured into three layers of complexity [13]: Starting from dynamical models over maneuver-based to complex interaction-aware models. The majority of trackers (e.g. [7, 11, 14, 25]) use Kalman filters that model motion using basic physical state dynamics. Maneuver-based methods (e.g. [2, 10, 19, 28]) find patterns in previously observed trajectories to predict the future evolution of the trajectory. Many current maneuver-based models focus mostly on a single class of objects, e.g. vehicles or pedestrians. Interaction-aware approaches (e.g. [12]) additionally take scene context into account, e.g. information about road crossings or relative positions of other vehicles. Typically, previous interaction-aware approaches have been limited to a small variety of specific scenes due to the high complexity of acquiring training data and modeling scene variations. Our method is maneuver-based with the capability of handling a variety of objects. Compared to previous maneuver-based methods, we additionally distinguish the observed objects by their shape which allows us to assign object category-specific motion patterns.

While most related work focuses on supervised methods (e.g. [17, 21]), only a small number of approaches tackle the semi-/unsupervised categorization of objects [22, 25]. Teichman and Thrun [23] propose a semi-supervised approach to classifying objects in street scenes into a set of given categories. Their approach is based on laser data, which is more accurate than the stereo data used in our method. They combine shape and appearance features to classify objects [25] in single frames of combined laser and vision data. Tao *et al.* [22] propose a semi-supervised learning method to classify 3D laser measurements and RGB-D images of objects into a set of given categories. The method represents shapes by hierarchical matching pursuit features [1] on reprojected depth images, and employs a variant of online star clustering [8] to build up a bipartite graph clustering of the shapes. In contrast to our approach, this method does not explicitly take trajectory information into account. Luber *et al.* [15] cluster moving objects into categories based on their time-varying shape. They use 2D laser data in a horizontal plane to detect and track objects, and to describe their 2D shape. The primary goal of this method is to provide a probabilistic model for classification of newly observed laser tracks. Our method observes shape from noisy 3D stereo data and clusters objects based on shape and additionally trajectory information. Compared to Luber *et al.* [15] we provide probabilistic SMPs that allow for predicting possible future trajectories even for static objects.

Several approaches have been proposed that learn motion models of objects in order to recognize events or predict their future motion. The early work of Johnson and Hogg [9] models the probability density functions of observed 2D image trajectories of tracked pedestrians using vector quantisation. Training examples are labelled by event types and newly observed trajectories are classified via nearest neighbors. Joseph *et al.* [10] learn a Bayesian non-parametric model of GPS position trajectories. Kooij [12] propose to use a Dirichlet process prior to model a mixture of linear dynamic systems. Their system is specifically designed for tracking persons in camera images. In contrast to the above methods, our method exploits 3D shape to further distinguish a variety of objects and their motion behaviour. By this, we also improve motion prediction.

3 Hierarchical Motion Model Clustering

We propose to describe objects in a hierarchical approach, as visualized in Fig. 2. In the following, we explain the steps in detail.

Training Phase: In a training phase, we learn SMPs from tracked objects.

1. First, we run a tracker on the training set. Different kinds of trackers can be used, e.g. detection-based or generic trackers. We solely require tracks with position estimates and 3D segments. By sampling from the tracks, a training set is gathered. A training example contains a shape model of the object, together with a sub-trajectory (see Sec. 4).
2. We cluster the instances from the training set based on their shape using Affinity Propagation (AP, [4]). The shape clusters $c_S \in \mathcal{C}_S$ represent categories of objects, also differentiating viewpoints within an object class. We choose AP as clustering algorithm

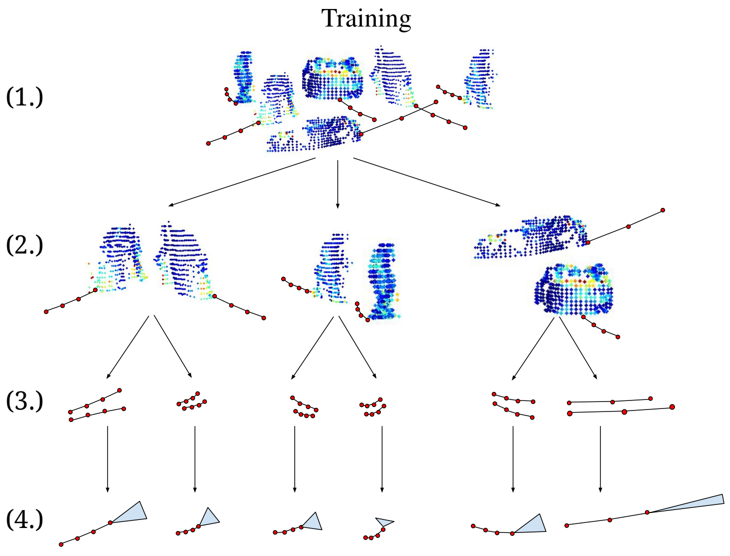


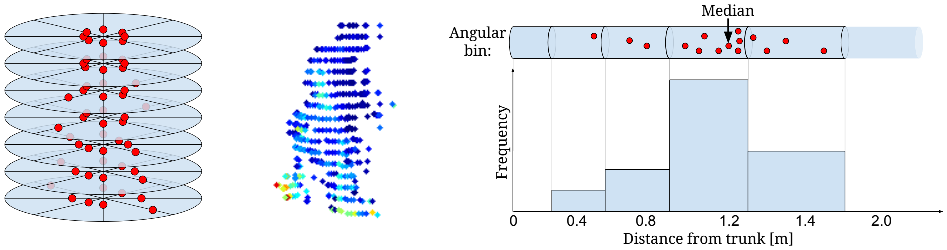
Figure 2: Motion prediction pipeline based on hierarchical clustering. Steps enumerated as in Sec. 3.

because it automatically estimates the number of clusters and provides cluster representatives, which we will need in the recall phase as cluster exemplars. Furthermore, we desire objects which are fully observed as cluster representatives. With the preference parameter of AP we can favour fully observed objects proportionally to the degree of observation. We optimize the parameters of AP by minimizing the prediction error (see supplementary material).

3. Various instances in each shape cluster can have different motion models, e.g. a car can be parked or drive with a high velocity. Hence, we cluster the trajectories $m \in \mathcal{M}(c_S)$ of each shape cluster c_S using AP to obtain shape-specific trajectory clusters $c_{M|c_S} \in \mathcal{C}_{M|c_S}$.
4. Each trajectory cluster in a shape forms one SMP $p = \left(c_S, c_{M|c_S}\right)$. The trajectories in the SMP are described by a Gaussian distribution on the past and future positions relative to their current position.

Recall Phase: Using our learned model, we classify novel object shapes and trajectories into one of the learned SMPs in the following way (see Sec. 5):

5. We gather shape and trajectory information from the same kind of tracker that we use in the training phase. This gives us shape information combined with trajectory information. However, in contrast to the training phase only the past trajectory is available.
6. By comparing the integrated shape of the new instance to all cluster centers we extract a subset of shape centers which are most similar to the observed object (see Sec. 5.1).
7. We compare all motion models from this subset of shape clusters to the observed trajectory and select the most similar one.
8. Based on the selected motion model we predict the future motion (see Sec. 5.2). Furthermore, we use the subset of motion models from the previous step to infer a probability distribution describing where the object could move based on its shape.



(a) Layer structure

(b) GCT of a walking pedestrian

(c) The content of an orientation bin can be represented by a median or a histogram

Figure 3: Schematical structure of a Generalized Christmas Tree.

4 Shape and Motion Features

4.1 Shape Features

We choose Generalized Christmas Trees (GCTs [17]) as shape representation. GCTs allow for aligning and integrating individual frame stereo reconstructions on the object over multiple frames, which smoothes out noise and enables us to describe shape variation in non-rigid parts. They provide a fixed-dimensional representation irrespective of the object shape and size. Recently, it was demonstrated that GCTs can be used as shape descriptors for various machine learning tasks such as classification, orientation and body pose estimation [17].

As visualized in Fig. 3 (a), a GCT consists of several height layers, which are further subdivided into angular bins around the vertical axis. Each angular bin maintains statistics on the distance of the observed 3D points on this part of the object surface to the GCT center axis (Fig. 3 (b) and (c)). In order to improve robustness, we adapt the center during tracking (see supplementary material for details). We represent these statistics either by the median or by a histogram of the points. The median-based representation is very well suited for rigid objects, while variations caused by non-rigidness are better captured using the histogram representation. To achieve a high granularity for thin objects and to increase size-invariance, we use a log-polar representation.

We do not assume that tracking directly provides a canonical orientation for the objects. Instead, we propose to extract a repeatable orientation estimate from the GCTs in order to make the shape features invariant to the object's orientation and to align the trajectories within a shape cluster. Note that the estimated orientation does not need to be an interpretable orientation, e.g. the object's moving or viewing direction. It only needs to be consistent across similar shapes and trajectories. Since we also want to represent static objects, we do not take the trajectory into account. We robustly determine the dominant plane within the GCT shape model using RANSAC [3] and use its normal as orientation estimate (see supplementary material for details). Ambiguities in the dominant plane are handled by including multiple possible orientations during training.

More formally, a GCT yields for each height layer $i \in \{1,\ldots,N\}$ and orientation bin $j \in \{1,\ldots,M\}$ either a median distance $s(i,j) \in \mathbb{R}$ or a B-dimensional histogram $s(i,j) \in \mathbb{R}^B$ of distance values. We make the GCT orientation-invariant by aligning the first orientation bin with the dominant orientation. We compare two GCTs s and s' by the average distance

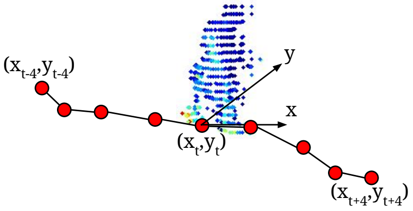


Figure 4: Visualization of tracklets. Red dots visualize object positions.

of the orientation bins [17], where $d(x,y)$ denotes the distance between bins x and y,

$$d(s,s') = \frac{1}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M} d(s(i,j),s'(i,j)). \tag{1}$$

Mitzel *et al.* [17] propose to compare median-based GCTs using the Euclidean distance and histogram-based GCTs using histogram intersection. Note that the objects are typically only partially observed, such that larger parts of the GCTs are not filled with observed values. Using zero values as default would strongly affect the distance measure between similar shapes, if the partially observed region does not coincide fully. Hence, empty, i.e. unobserved, bins are filled with a default median distance or default histogram, respectively. The optimal value for the default distance or histogram is estimated on a training dataset. We include further details on distance measures and default values or histograms in the supplementary material.

4.2 Trajectory Features

Motion is represented by sub-trajectories $m = (x_{t-T}, y_{t-T}, \ldots, x_{t+T}, y_{t+T})$, which we denote as tracklets. They range within a fixed interval of past and future frames T around each current frame at time t. Tracklets include the projected xy-positions on the ground plane at associated time steps. As suggested in [28], this simple representation can model all types of motion, even ones that cannot be described using constant velocity or acceleration models.

In order to describe the tracklet in an object-centric and orientation-invariant way, we express the tracklet positions in a local object coordinate frame. The origin of this frame is placed at the current center position of the tracklet and aligns the tracklet's x-axis with the orientation of the object shape as estimated in Sec. 4.1. For AP clustering, we determine the Euclidean distance of tracklet positions that correspond by time steps.

5 Prediction

5.1 Motion Model Lookup

To match a motion model to a newly observed object instance with shape feature s and trajectory feature m, we first find the subset $\mathcal{C}_S(s)$ of the shape clusters $\mathcal{C}_S$ which contain the most similar shapes to s,

$$\mathcal{C}_S(s) := \left\{ c_S \in \mathcal{C}_S \,\middle|\, d(s,\widehat{s}(c_S)) < \lambda \min_{c'_S \in \mathcal{C}_S} d(s,\widehat{s}(c'_S)) \right\}, \tag{2}$$

where $\widehat{s}(c_S)$ is the exemplar shape of cluster c_S and parameter λ controls the number of closest shape clusters.

Among the trajectory clusters $\mathcal{C}_M(s)$ contained in the shape clusters $\mathcal{C}_S(s)$, we find the best matching motion cluster $\widehat{c}_M$ with respect to the observed trajectory $m_{t-T:t}$ from the T past time frames by

$$\widehat{c}_M(s, m_{t-T:t}) = \arg\min_{c'_M \in \mathcal{C}_M(s)} \left\| m_{t-T:t} - \widehat{m}_{t-T:t}(c'_M) \right\|_2, \tag{3}$$

where $\widehat{m}(c_M)$ is the exemplar trajectory of cluster c_M.

While we chose only complete sub-trajectories for our training data the trajectories of the newly observed object instance can be shorter. In that case we shorten the trajectory models to the length of the trajectory of the newly observed object.

5.2 Motion Prediction

Based on the selected trajectory cluster $\widehat{c}_M(s, m_{t-T:t})$, we predict the future motion of the object. We model the motion distribution within the trajectory cluster as Gaussian in the *xy*-positions per time step with mean $\mu_{t-T:t+T}$ and covariance $\Sigma_{t-T:t+T}$. The mean future motion of the observed object is then obtained by conditioning on the past observed trajectory,

$$\mu_{a|b} = \mu_a + \Sigma_{ab} \Sigma_{bb}^{-1} (m_b - \mu_b), \tag{4}$$

where we used the shorthands $a := t+1 : t+T$ and $b := t-T : t$. If only part of the past trajectory is observable, we can condition on this shorter history as well through proper Gaussian marginalization.

6 Experiments

We evaluate our approach qualitatively and quantitatively on the KITTI tracking dataset [6]. This dataset contains 21 training sequences recorded with a front-facing stereo camera on a driving car in urban street scenes. We use the training sequences, since the images are annotated with ground truth object bounding boxes (e.g. cars, vans, trucks, cyclists, pedestrians) and their correspondence between frames.

We employ two kinds of trackers for our evaluation: An "oracle" tracker uses the annotated bounding boxes and correspondences as object observations within a multi-object Kalman filter tracking framework. The Kalman filter propagates the state with a constant-velocity motion model. Stereo depth [8] is used to segment and localize the objects. These 3D shape measurements are aligned and integrated into GCT shape models over time [17]. The second tracker is a tracking-by-detection approach that extends the QPBO tracker in [14]. We use object detections for cars, pedestrians and cyclists [27] and fuse these detections with 3D region proposals [18], which provides a 3D shape segmentation of the detected objects. We associate detections and proposals based on spatial proximity and image-domain bounding-box overlap. The segmented shapes are aligned and integrated into GCT models.

In the following, we analyze the clustering achieved by our unsupervised learning method, and assess the accuracy of our motion prediction approach.

6.1 Qualitative Feature Evaluation

We use the oracle tracker to obtain shape features and embed the features with t-SNE [26] to visualize their expressiveness. The left visualization in Fig. 5 shows a t-SNE embedding

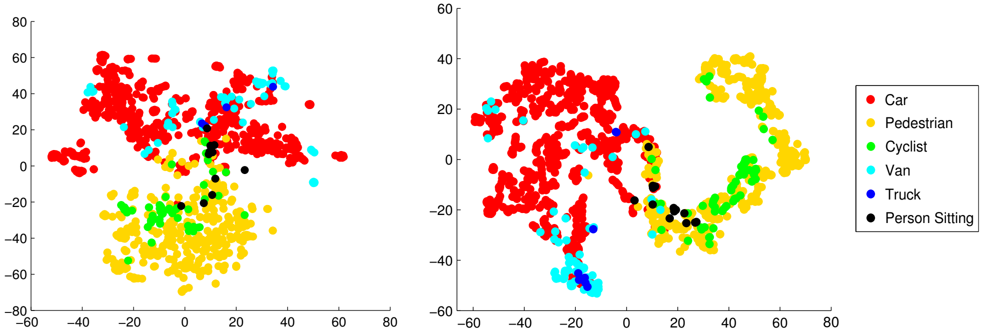


Figure 5: t-SNE embedding of the feature space. Histogram-based GCTs without default values *(left)* and with default values *(right)*.

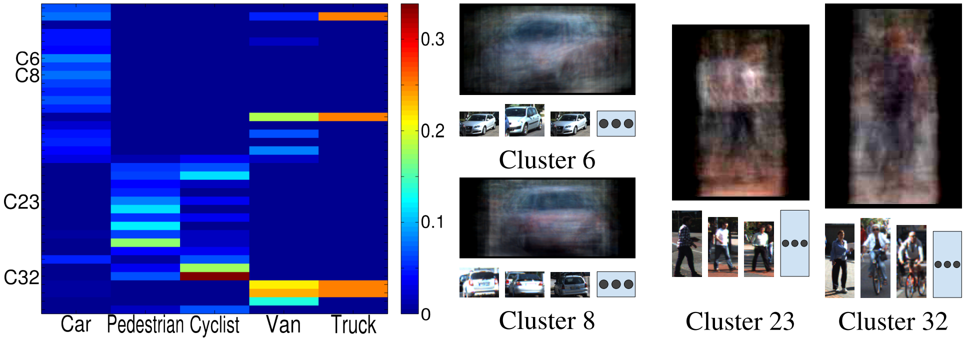


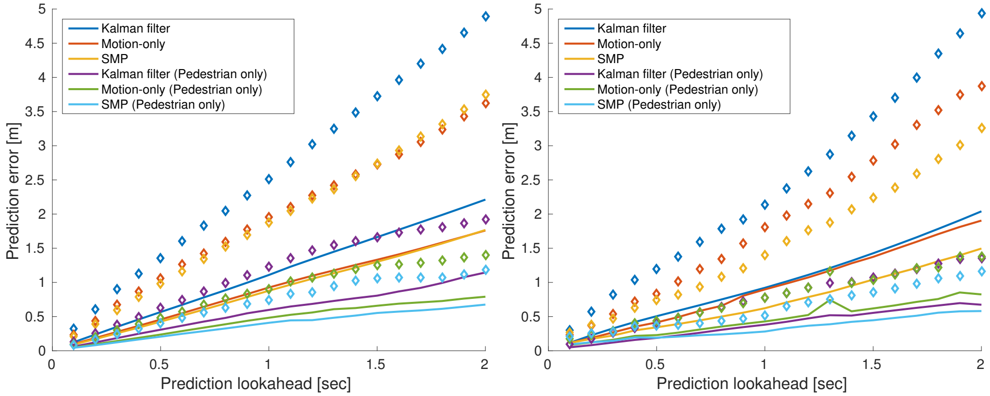
Figure 6: Confusion matrix (column-normalized) and mean images of a shape clustering.

of the shape features represented as histograms. It indicates that objects of different size can be kept apart using the shape features. The distinction between similar sized objects, e.g. pedestrians and cyclists, is not that strong but recognizable.

Fig. 5 demonstrates the importance of using appropriate default values for the unobserved parts in the GCTs. We can see that using default values, the instances belonging to the same object class get mapped closer to each others than without default values. Fig. 6 shows the confusion matrix between the labelled categories and the clusters generated through AP on shape features. First, we can see again that large and small objects are separated well, while objects of similar size and shape are mixed. We also visualize the categorization of the clusters with the mean of the image patches within a cluster. Clusters 6 and 8 are both homogeneous car clusters but vary in viewpoint. Clusters 23 and 32 contain mainly pedestrians and cyclists, but the clusters are from different viewpoints. Furthermore, the mean image of cluster 32 reveals a similarity between pedestrians and cyclists: They can appear similar from the front.

6.2 Motion Prediction

We evaluate the accuracy of our SMP-based motion prediction approach on the KITTI tracking sequences. We represent the shapes with histogram-based GCTs and use histogram intersection as distance measure [16]. For comparison, we also provide results with a “motion-only“, maneuver-based approach that performs AP clustering just on the trajectories, but that also uses the motion prediction approach described in Sec. 5. It matches trajectories



(a) Oracle tracker (b) Detection-based tracker

Figure 7: Long-term prediction error depending on the prediction time.

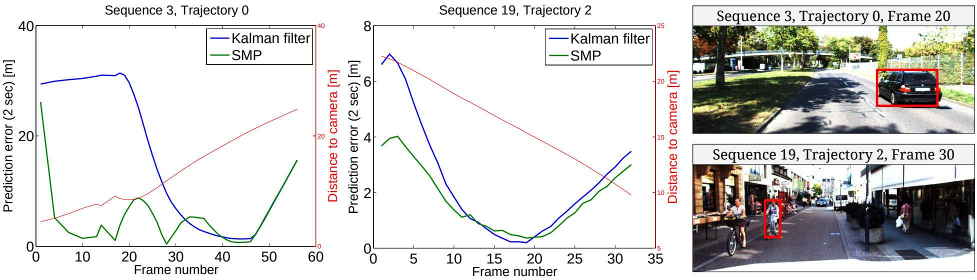


Figure 8: Prediction error for two example trajectories.

directly to the trajectory clusters without regarding shapes. We perform leave-one-out cross-validation and provide average results over the validations.

Fig. 7 visualizes the performance of our pipeline with a lookahead of up to 2 seconds ($T = 20$) which is regarded as long-term prediction by [20]. The diamonds denote the 90% quantile. In Fig. 7 (a), we use the oracle tracker to train our SMP and the motion-only clustering approaches. As evaluation we compare the Euclidean distance between the predicted positions and the tracker output. Hence, in this experiment the methods learn to imitate the estimates of the oracle tracker. We see that the SMP-based and the motion-only approach perform better than the Kalman filter. We further observe that reasoning on shape increases performance, especially for the less frequent object categories like pedestrians whose trajectory models differ significantly from the shape-independent trajectory models. Note that while the Kalman filter is limited by its motion model to linear extrapolations, both learning approaches can use the trajectory history to predict more complex future motion, *e.g*. when an object is moving along a curve. The SMP approach clusters the available trajectories within a shape cluster, such that it has less data available to find trajectory clusters. The motion-only approach can transfer a trajectory model learned from a car to a cyclist which is beneficial for less data but biases the motion models.

Fig. 8 shows the prediction error for two moving objects along their track (frame 0 means initialization frame). The left example shows the error for a car which overtakes on the right with a high speed (traj. 0, seq. 3). The right example is a cyclist that drives towards the camera (traj. 2, seq. 19). The Kalman filter requires several frames to initialize its position and velocity estimate for prediction. Our method can quickly obtain a shape model, from which it seems to infer predictions better in both examples.

In Fig. 7 (b), we train SMPs and the motion-only model in an unsupervised way using the tracking-by-detection Kalman filter tracker. In this experiment we evaluate against the annotated ground-truth. The SMP-based prediction deviates by 1.49m from the ground-truth while the motion-only prediction deviates by 1.90m. We attribute this to the capability of SMPs to capture shape and trajectory outliers in dedicated clusters.

The deviation between Kalman filter and SMP predictions is only about 20cm for pedestrians on a 2s lookahead. We assume this is due to the fact that the observed motions are quite consistent and fairly linear.

Building clusters from example data consisting of 2000 sub-trajectories takes several minutes on our machine (quad core with 3.4GHz, 16GB memory). To enrich the data further we propose to learn incrementally, e.g. with group induction [24]. Predicting motion based on a given model requires $|\mathcal{C}_S|$ shape distance comparison to find the closest clusters and it requires $|\mathcal{C}_M(s)|$ trajectory distance comparisons for each selected shape cluster. Since the number of clusters is small, this process takes only few milliseconds.

7 Conclusion

In this paper, we propose an unsupervised learning approach that categorizes objects in urban street scenes based on their shape and motion. We use a two-layer hierarchical clustering process, in which we first distinguish objects by shape, and then cluster the trajectories within each shape cluster, forming shape-motion patterns (SMPs). Our approach uses stereo vision, which provides noisy and partial 3D reconstructions of the objects. These reconstructions are aligned and fused in GCT shape models in order to smooth out noise and increase the level of completeness of the shape model. The motion of the objects is tracked using a Kalman filter to obtain tracklets. Lastly, we propose a method to predict the motion of objects based on their shape and motion similarity with learned SMPs.

In experiments, we analyse the suitability of our shape features for the clustering task. We also assess the accuracy of our SMP-based motion prediction and compare our method with predictions based on motion-only clustering and Kalman filters. Our evaluation demonstrates that SMPs can outperform motion-only and Kalman filter prediction.

In contrast to a Kalman filter based tracker, our method can provide a versatile set of learned motion patterns for a recognized shape, which can be used to assess potential future motions of objects. Our method is not restricted to moving objects in contrast to pure trajectory-based methods. It can also provide potential motions based on the shape, even if no prior information on the motion of the object is available or the object is static.

In future work, we plan to explore the use of further cues such as scene context to further distinguish the behavior of objects. Our method could also extend single-class maneuver-based trackers with motion models for a larger variety of objects.

Acknowledgments: This work was funded by ERC Starting Grant project CV-SUPER (ERC-2012-StG-307432). We would like to thank Wolfgang Mehner for helpful discussions.

References

[1] Liefeng Bo, Xiaofeng Ren, and Dieter Fox. Unsupervised feature learning for RGB-D based object recognition. In *ISER*, 2012.

[2] Simone Calderara, Andrea Prati, and Rita Cucchiara. Mixtures of von mises distributions for people trajectory shape analysis. *IEEE Trans. on Circuits and Systems for Video Technology*, 21(4):457–471, 2011.

[3] Martin A. Fischler and Robert C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Comm. ACM*, 24(6):381–395, 1981.

[4] Brendan J Frey and Delbert Dueck. Clustering by passing messages between data points. *Science*, 315(5814):972–976, 2007.

[5] Andreas Geiger, Martin Roser, and Raquel Urtasun. Efficient Large-Scale Stereo Matching. In *ACCV*, 2010.

[6] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In *CVPR*, 2012.

[7] David Held, Jesse Levinson, Sebastian Thrun, and Silvio Savarese. Combining 3D Shape, Color, and Motion for Robust Anytime Tracking. In *RSS*, 2014.

[8] Aslam A. Javed, Ekaterina Pelekhov, and Daniela Rus. The star clustering algorithm for static and dynamic information organization. *Journal of Graph Algorithms and Applications*, 8:95–129, 2004.

[9] Neil Johnson and David Hogg. Learning the distribution of object trajectories for event recognition. *Image and Vision Computing*, 14:583–592, 1996.

[10] Joshua Joseph, Finale Doshi-Velez, Albert S Huang, and Nicholas Roy. A bayesian nonparametric approach to modeling motion patterns. *Autonomous Robots*, 31(4):383–400, 2011.

[11] Ralf Kaestner, Jérôme Maye, Yves Pilat, and Roland Siegwart. Generative Object Detection and Tracking in 3D Range Data. In *ICRA*, 2012.

[12] Julian Kooij. *Generative Models for Pedestrian Track Analysis*. Ridderprint BV, 2015.

[13] Stéphanie Lefèvre, Dizan Vasquez, and Christian Laugier. A survey on motion prediction and risk assessment for intelligent vehicles. *Robomech Journal*, 1(1), 2014.

[14] Bastian Leibe, Konrad Schindler, Nico Cornelis, and Luc Van Gool. Coupled Object Detection and Tracking from Static Cameras and Moving Vehicles. *PAMI*, 30(10): 1683–1698, 2008.

[15] Matthias Luber, Kai O Arras, Christian Plagemann, and Wolfram Burgard. Classifying dynamic objects. *Autonomous Robots*, 26(2-3):141–151, 2009.

[16] Dennis Mitzel and Bastian Leibe. Taking Mobile Multi-Object Tracking to the Next Level: People, Unknown Objects, and Carried Items. In *ECCV*, 2012.

[17] Dennis Mitzel, Jasper Diesel, Aljoša Ošep, Umer Rafi, and Bastian Leibe. A fixed-dimensional 3d shape representation for matching partially observed objects in street scenes. In *ICRA*, 2015.

[18] Aljoša Ošep, Alexander Hermans, Francis Engelmann, Dirk Klostermann, Markus Mathias, and Bastian Leibe. Multi-scale object candidates for generic object tracking in street scenes. In *ICRA*, 2016.

[19] Victor Romero-Cano, Juan Nieto, Gabriel Agamennoni, et al. Unsupervised motion learning from a moving platform. In *Intel. Vehicles Symp.*, 2013.

[20] Sayanan Sivaraman and Mohan Manubhai Trivedi. Looking at vehicles on the road: A survey of vision-based vehicle detection, tracking, and behavior analysis. *IEEETITS*, pages 1773–1795, 2013.

[21] Luciano Spinello, Rudolph Triebel, and Roland Siegwart. Multiclass multimodal detection and tracking in urban environments. *IJRR*, 29(12):1498–1515, 2010.

[22] Ye Tao, Rudolph Triebel, and Daniel Cremers. Semi-supervised online learning for efficient classification of objects in 3d data streams. In *IROS*, 2015.

[23] Alex Teichman and Sebastian Thrun. Tracking-based semi-supervised learning. *IJRR*, pages 804–818, 2012.

[24] Alex Teichman and Sebastian Thrun. Group induction. In *IROS*, 2013.

[25] Alex Teichman, Jesse Levinson, and Sebastian Thrun. Towards 3d object recognition via classification of arbitrary object tracks. In *ICRA*, 2011.

[26] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *JMLR*, 9 (Nov):85, 2008.

[27] Xiaoyu Wang, Ming Yang, Shenghuo Zhu, and Yuanqing Lin. Regionlets for generic object detection. In *ICCV*, 2013.

[28] Jürgen Wiest, Matthias Höffken, Ulrich Kresel, and Klaus Dietmayer. Probabilistic trajectory prediction with gaussian mixture models. In *Intel. Vehicles Symp.*, pages 141–146, 2012.