Monocular Arbitrary Moving Object Discovery and Segmentation

Michal Neoral
neoramic@fel.cvut.cz
Jan Šochman
jan.sochman@fel.cvut.cz
Jiří Matas
matas@cmp.felk.cvut.cz

Czech Technical University in Prague
Faculty of Electrical Engineering
Department of Cybernetics

Abstract

We propose a method for discovery and segmentation of objects that are, or their parts are, independently moving in the scene. Given three monocular video frames, the method outputs semantically meaningful regions, i.e. regions corresponding to the whole object, even when only a part of it moves.

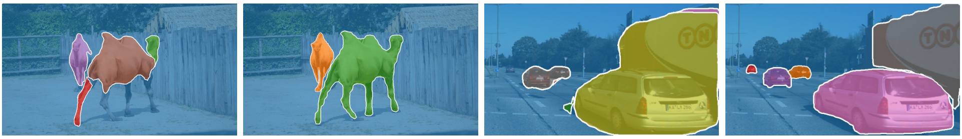
The architecture of the CNN-based end-to-end method, called Raptor, combines semantic and motion backbones, which pass their outputs to a final region segmentation network. The semantic backbone is trained in a class-agnostic manner in order to generalise to object classes beyond the training data. The core of the motion branch is a geometrical cost volume computed from optical flow, optical expansion, mono-depth and the estimated camera motion.

Evaluation of the proposed architecture on the instance motion segmentation and binary moving-static segmentation problems on KITTI, DAVIS-Moving and YTVOS-Moving datasets shows that the proposed method achieves state-of-the-art results on all the datasets and is able to generalise well to various environments. For the KITTI dataset, we provide an upgraded instance motion segmentation annotation which covers *all* moving objects. Dataset, code and models are available on the github project page github.com/michalneoral/Raptor.

1 Introduction

Segmenting a dynamic scene into independently moving parts is a practical task with a wide range of applications like video editing, autonomous driving or human-robot interaction. In this paper we focus on an important subset of the problem, where only a single camera is available (monocular vision), the camera is not necessarily static and the processing is required to be causal. The goal is to discover and segment *all* independently moving "objects". We refer to the problem as *moving object discovery and segmentation* – MODaS.

Object definition. The crux of the MODaS specification is the definition of the instance or the object. The commonly used definition (see Section 2) is geometrical – a group of pixels which undergo the same rigid motion and are connected spatially. This definition



(a1) Raptor (MO,FW) (a2) **Raptor** (b1) Raptor (MO,FW) (b2) **Raptor**

Figure 1: Failures of motion-only "object" definition. (a1) Non-rigid objects are over-segmented and static parts missed with Raptor (MO,FW), i.e without the semantic branch. (b1) Overlapping semantically meaningful objects with the same motion returned as one component. In both cases, the proposed method, Raptor, produces the desired output (a2, b2).

fails in two important cases: (i) when the object is not rigid, and (ii) when two semantically different but nearby objects move similarly (see Figure 1). For applications, both the over- and under-segmentation cases are problematic. For instance, when a person moves her leg, the entire person outline should be returned as it is the person whose position is going to be predicted, not the leg. The moving object is thus defined both geometrically, its part moves independent of the camera motion, and semantically, it is a semantically meaningful entity.

Object discovery is closely related to modern detectors, as they output not just the position, but also a segmentation of possibly hundreds of classes [18]. In this respect we go one step further and use a detector trained in a class-agnostic way [14], where all classes are merged into a single "object" class. This way, the detector generalises better to objects of classes not seen during the training (*e.g.* a camel is detected when the training data contain only horses). To distinguish it from the detection problem with known classes, we call it object *discovery*.

Independent motion. An object (or its part) is considered moving independently when its apparent motion is not a consequence of camera ego-motion. Recognising such motions from a monocular camera is an ill-posed problem in general. As the optical flow itself is only a 2D projection of the 3D scene motion, its interpretation is ambiguous. Recent advances in the monocular depth estimation offer a possible way to overcome these ambiguities. For scenes with statistics similar to the training dataset, the mono-depth serves as a useful prior of the true depth and is able to disambiguate the observed motion [46].

Contributions. We propose a CNN-based architecture called Raptor, based on a novel combination of semantic and geometrical processing. We show that the class-agnostic semantic part leads to discovery of semantically meaningful objects, while the geometrical motion cost volume processing resolves the apparent motion ambiguities. The network discovers both rigid and non-rigid moving objects and their instance segmentation masks. Unlike the most methods [14, 15, 20, 31, 33, 38, 46], the Raptor architecture uses three frames for the MODaS. We are the first to extend the geometrical part beyond two frame processing. We show that estimation of geometrical features in both directions (forward and backward) increases the precision of both the discovery and the segmentation outputs of the Raptor.

The method was evaluated on three standard benchmarks: DAVIS-moving [14, 28], KITTI [23], and YTVOS-Moving [14, 43] and it achieved state of the art results, often significantly surpassing other methods. The evaluation used an extended set of metrics and Raptor is the top performer in all of them. Strikingly, the excellent performance of Raptor was achieved despite being trained only[1] on the COCO instance segmentation dataset [21] and the synthetic FlyingThings3D dataset [22]. The diversity of the evaluation benchmark

[1] Individual modules used for motion cost volume computation were trained on different datasets.

data and their difference from the training set makes it likely that Raptor will generalise well to new environments.

As an additional minor contribution, for the KITTI dataset, we extended the official independently moving instance segmentation ground truth to cover all moving objects, not just a selection of cars and vans. We present results for a number of state-of-the-art methods on this updated dataset.

2 State of the Art

The first monocular motion segmentation algorithms appeared already in the 90's. Some of them [4, 13, 19, 40] used robust but typically hand-crafted algorithms to separate the optical flow into 'layers' modelled by an affine motion. The problem was later given a more formal treatment in [34]. In [19], motion segmentation was also already connected with tracking. At about the same time, the first optimisation approaches using normalised graph cut were proposed [30]. The focus of these approaches was usually the robustness to the imperfect optical flow and the assumption of a simple geometrical motion model for the objects and the scene. During this period, the first multi-body factorisation methods started to appear as well [5, 12]

In the next decade, approaches built on Bayesian treatment of the problem [3], improved on the multi-body factorisation [37], or proposed to integrate the motion segmentation into the variational formulation of the optical flow estimation using level sets [8]. Going beyond two frames only, feature trajectories clustering approaches appeared [7, 44] using geometry and locality properties of the trajectories. Some approaches started to cluster pixels based on simple appearance cues and region detectors [9, 10]. To reduce the computation, the frames were also over-segmented into super-pixels [1].

Although there had existed checker-board multi-body factorisation datasets like Hopkins 155 [35], the first real-world dataset for evaluation was the BMS-26 dataset [7], which was later extended into the FBMS-59 dataset [26].

Before the advent of the CNN approaches, a few more notable approaches appeared, mostly re-iterating and improving on the previous ideas: *e.g.* detachable objects [2], more advanced trajectory segmentation approaches [16, 25, 26], or a CRF-based approach [36]. Probably thanks to the FBMS dataset, the focus was mostly on the video segmentation. Later, a few more datasets were introduced, most of them not directly for the MODaS problem, but for related tasks [17, 27, 42]. All the above approaches focus on the geometrical aspect of the task and completely ignore the semantic aspect. The approaches cannot compete with the speed, robustness and performance of the modern CNN-based approaches.

The first CNN-based attempts explored their ability to learn the geometric segmentation in a fully convolutional way [33], used semantic object detector for video segmentation proposal generation [15], but also introduced a two stream architecture with one branch capturing the appearance and the other motion features, though applied to general object detection only [20]. The SfM-Net approach [38] even attempted to train a do-it-all partially unsupervised CNN. A motion segmentation MODNet [31] introduced a similar two-branch architecture with two heads, one for bounding box regression and the other for segmentation. However, the detection head ignored the motion cues altogether. In [47], a monocular depth, ego-motion estimation and corresponding inlier mask are also integrated into a purely geometrical and partially unsupervised pipeline. An alternative strategy is to train the trajectory embedding using an RNN and use the trajectory clustering approach as before [41].

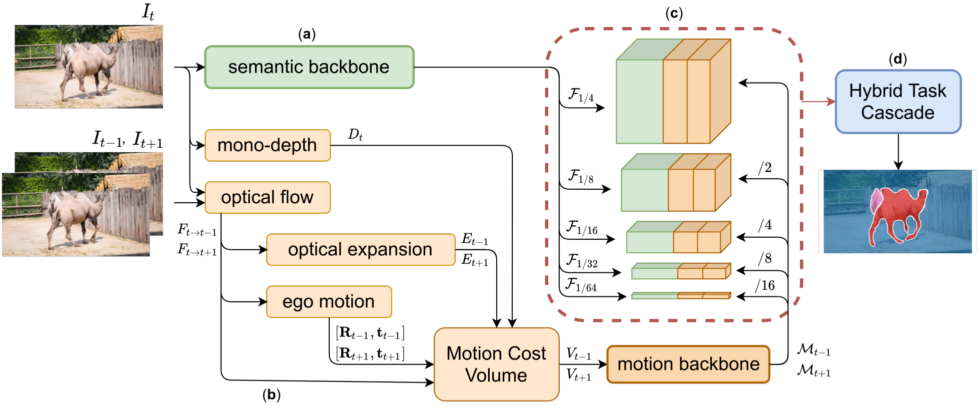


Figure 2: Raptor architecture components: (**a**) *the semantic branch* (green), pre-trained on the class-agnostic COCO dataset, outputs deep features for the current frame I_t, (**b**) the siamese *motion branch* (brownish) which produces the motion cost volume using several geometric transformations of the three input frames, (**c**) the concatenated outputs of (a) and (b), (**d**) the *Hybrid Task Cascade* (blue), which estimates instance motion segmentation from (c).

The most relevant approach to ours is that of [14]. Their architecture has two input streams followed by a region proposal network and two heads, one for bounding box prediction and the other for segmentation. The motion input branch takes the optical flow estimate as input and is trained to detect all moving objects. The appearance branch is trained in a class-agnostic way on the COCO dataset [21]. The main weakness of this architecture is the reliance on the optical flow only in the motion stream. It is known to be ambiguous to segment moving objects from a 2D projection of the true motion only.

The geometry ambiguities appearing in monocular motion segmentation were treated in [46]. For every ambiguity they propose a cost which handles that particular case. All the costs are packed into a motion cost volume which is passed to a rigid-body motion segmentation CNN. This purely geometrical approach leads to a state-of-the-art scene flow when the stereo depth is used, but struggles with the imperfect mono-depth. The motion segmentation works relatively well for rigid-body motion, but over-segments articulated objects.

We take inspiration in the last two approaches and propose a method that combines the strengths of both of them.

3 The Raptor Architecture

The proposed Raptor architecture is shown schematically in Figure 2. It has two input branches: one producing semantic features, green, the other producing motion features using the motion cost volume (MCV), orange. Their outputs are concatenated and sent to a Hybrid Task Cascade (HTC) [11] head which produces the moving objects masks.

The semantic backbone (Fig. 2 (**a**)) is adapted from the DetectoRS architecture [29]. It is built around a Recursive Feature Pyramid backbone (RFP) with Switchable Atrous Convolutions (SAC). We use the ResNet-50 variant of the backbone.

To transform the detection pipeline into an object discovery method so that it generalises better to unknown classes, we follow the idea of class-agnostic training of [14]. We train both the backbone and a temporary HTC head on an object segmentation problem [21] with all object categories merged into a single "object" category. The motion branch is not present for this training. It has been show in [14], that this way the method predicts better unknown classes. After this training, the HTC head is discarded and the semantic backbone weights are fixed.

The motion branch (Fig. 2 (**b**)) is inspired by a recent rigid-body motion segmentation approach [46]. It uses monocular depth D_t, optical flow $F_{t \to t+1}$, optical expansion E_{t+1}, and ego-motion estimate, $[\mathbf{R}_{t+1}, \mathbf{t}_{t+1}]$, as inputs for the forward motion cost volume (MCV) construction. We further compute the "backward" MCV using $F_{t \to t-1}$, E_{t-1} and $[\mathbf{R}_{t-1}, \mathbf{t}_{t-1}]$. As in [46], we build a fourteen[2] channel MCV for each direction consisting of: per-pixel Samson error of the epipolar geometry, per-pixel rotational homography re-projection error, 3D P+P cost, depth contrast cost, reconstructed 3D scene points from optical flow, rectified motion field, uncertainty of optical flow and optical expansion and 3D angular P+P cost. These costs are designed to indicate inconsistency with the estimated ego-motion while dealing with various ambiguities of co-planar or co-linear motion or the ego-motion degeneracy.

The motion branch backbone architecture is similar to the semantic backbone, but instead of ResNet-50 it builds on the ResNet-18 DetectoRS architecture. The input is a 14 channel motion cost volume for each direction. Image features do not input directly to the motion backbone. The MCVs for forwards and backwards directions pass-through motion backbone one by one. The motion branch is pre-trained separately with another temporary HTC head and without the semantic backbone on the MODaS problem. After this training, the temporary HTC head is discarded again and the motion backbone is fixed.

Semantic + motion (Fig. 2 (**b**)) The outputs from the two branches are concatenated. The semantic branch produces five feature tensors $\mathcal{F}_{1/4}$, $\mathcal{F}_{1/8}$, $\mathcal{F}_{1/16}$, $\mathcal{F}_{1/32}$, $\mathcal{F}_{1/64}$ with decreasing spatial resolution and each with 256 channels.

The original motion branch produced feature tensors with 256 channels for each direction. Facing the memory limitation of the training device, we normalise the outputs to 128-channels by an extra 1x1 convolution layer. Then, the outputs for both directions are concatenated with the features from semantic backbone and are fed to the HTC head (512 channels in total). As the MCV is designed for rigid body motion only, it reports part-only inconsistency for articulated objects. It is the task of the HTC head to combine these partial inconsistencies together with the semantic features and output the complete moving object segmentation masks.

Both branches are fixed for the final training and only the final HTC head is trained for moving object discovery. The motion branch normalisation 1x1 CNN layer is trained during the final stage of training together with HTC.

Improvement of MCV components. The motion branch uses several external algorithms to generate its inputs. We also took care to bring them to their most advanced versions available in literature. We use RAFT [32] instead of the original older and weaker VCN [45] optical flow estimator. We trained RAFT on a wide range of datasets (Robust Vision Challenge style as prior works [24, 39] demonstrate increased generalisation). Using this flow estimate we also re-trained the optical expansion part.

Unlike the original VCN, RAFT does not output the out-of-range confidence which is one of the channels in the MCV. We substitute it by a similar forward-backward consistency cost which is computed for MCV as $F_{t \to t+1} + \text{warp}(F_{t+1 \to t}, F_{t \to t+1})$, where the warp operation transforms the flow $F_{t+1 \to t}$ to the frame t. We do not threshold this value.

[2] Note the paper [46] describes 12 channels. The published code uses two extra channels.

4 Experiments

We first present Raptor preliminary settings and the training datasets. Then, we explain the evaluation metrics and compare Raptor with state-of-the-art instance motion segmentation algorithms. Finally, we present an ablation study of the Raptor configurations.

4.1 Preliminary Settings

Construction of the MCV requires intrinsic camera parameters and estimates of camera motion $[\mathbf{R}, \mathbf{t}]$ for the decomposition of essential matrices. We choose NG-RANSAC [6] for essential matrix estimation, since it allows possible end-to-end training in the future work. For the datasets which do not contain intrinsic camera parameters we set the focal length to $1/\max(I_{width}, I_{height})$ and the principal point is set to the middle point of the image.

4.2 Datasets

Training data. We trained the final stage of Raptor on the FlyingThings3D dataset [22] only. We follow the training procedure of RigidMask [46] for fair comparison and to demonstrate better generalisation over evaluated datasets. We computed instance motion segmentation masks from ground truth depth, optical flow, object masks and camera positions. While [46] already prepared such dataset for training, it is not publicly available. We made our training dataset public. We pre-train the class-agnostic segmentation branch of Raptor on the COCO object segmentation dataset [21].

Evaluation data. We test our model on the KITTI [23], DAVIS-Moving [14, 28] and YTVOS-Moving [14, 43] datasets. These datasets reflect a wide range of conditions for MODaS application, from autonomous driving (KITTI) to freehand camera motion (YTVOS, DAVIS). All of the datasets contain both rigid and non-rigid moving objects of a various semantic classes.

The original KITTI dataset [23] contains moving object segmentation ground truth, but it is restricted to a subset of moving cars and vans only. Other types of moving objects (buses, trains, pedestrians) are not part of the ground truth data. We manually labelled the sequences to complete the annotations. We refer to this extended motion segmentation dataset as *KITTI-MS+*. For compatibility with previous evaluations we show results on both KITTI and KITTI-MS+.

The KITTI dataset contains intrinsic camera parameters, which we use during the evaluation for camera motion estimation. DAVIS and YTVOS datasets do not contain the camera calibrations. Thus, we approximate them with the approach detailed in Section 4.1.

4.3 Metrics

To evaluate Raptor we consider all metrics used by previous methods and add a few more object-centric ones. We re-evaluate also the other methods using this extended measure set.

We adopt the standard *precision* (P), *recall* (R) and *F-measure* (F) introduced in [26]. However, these metrics do not penalise for false positive detections. Following [14] we thus evaluate methods also with updated *precision* (Pu), *recall* (Ru) and *F-measure* (Fu) metrics, which do penalise FPs.

KITTI'15

exp.	method	bg	obj	P	R	F	Pu	Ru	Fu	AO	FN	FP	AP
a	Rigid mask (mono) [46]	^{3}97.43	^{3}90.73	^{3}84.31	^{4}88.19	^{4}85.57	^{3}68.73	^{4}89.41	^{3}74.47	^{4}86.66	^{2}6.03	^{6}400	^{7}24.30
a	Towards [14]	^{6}96.19	^{6}84.47	^{5}78.41	^{6}85.05	^{5}81.29	^{5}67.74	^{6}84.38	^{5}70.97	^{5}85.55	^{6}29.47	^{3}155	^{5}31.80
a	Towards w/o tracking [14]	^{7}94.58	^{7}79.79	^{7}74.12	^{7}79.68	^{7}76.43	^{6}59.64	^{7}82.00	^{7}64.51	^{7}80.62	^{7}36.43	^{5}194	^{6}26.00
a+b	**Raptor**	^{1}98.40	^{1}94.33	^{2}84.90	^{1}94.09	^{2}88.93	^{2}78.15	^{2}92.30	^{1}82.18	^{1}92.46	^{4}18.56	^{1}95	^{2}54.30
b	Raptor (SEM)	^{5}96.44	^{4}89.77	^{1}88.12	^{2}93.78	^{1}90.40	^{7}58.24	^{1}95.00	^{6}67.69	^{2}92.18	^{1}1.16	^{7}1106	^{1}55.30
b	Raptor (MO, FW)	^{4}97.18	^{5}88.26	^{6}77.16	^{5}86.27	^{6}80.80	^{4}68.41	^{5}86.37	^{4}73.59	^{6}84.19	^{5}24.36	^{2}133	^{4}41.90
b	Raptor (SEM+MO, FW)	^{2}98.28	^{2}93.17	^{4}82.70	^{3}91.60	^{3}86.56	^{1}78.17	^{3}91.29	^{2}81.56	^{3}89.97	^{3}17.63	^{4}162	^{3}52.30

KITTI'15-MS+

exp.	method	bg	obj	P	R	F	Pu	Ru	Fu	AO	FN	FP	AP
a	Rigid mask (mono) [46]	^{3}98.39	^{3}79.32	^{6}76.56	^{6}75.29	^{6}75.13	^{3}79.75	^{4}83.82	^{3}79.32	^{7}68.28	^{2}40.66	^{6}284	^{6}20.00
a	Towards [14]	^{5}97.51	^{6}74.73	^{4}80.77	^{4}82.06	^{3}81.17	^{5}77.46	^{6}78.95	^{6}74.73	^{3}78.28	^{6}59.23	^{3}101	^{5}24.50
a	Towards w/o tracking [14]	^{7}96.71	^{7}69.13	^{7}73.92	^{7}74.73	^{7}74.00	^{7}69.57	^{7}76.99	^{7}69.13	^{6}71.80	^{7}63.33	^{5}146	^{6}20.00
a+b	**Raptor**	^{1}99.07	^{1}86.82	^{1}90.03	^{1}93.79	^{1}91.61	^{1}90.06	^{2}86.58	^{1}86.82	^{1}87.18	^{3}53.08	^{1}34	^{2}41.90
b	Raptor (SEM)	^{6}96.96	^{5}76.31	^{3}81.90	^{5}80.83	^{4}80.73	^{6}71.05	^{1}92.70	^{5}76.31	^{4}75.44	^{1}24.37	^{7}868	^{1}46.80
b	Raptor (MO, FW)	^{4}98.18	^{4}78.16	^{5}79.20	^{3}83.50	^{5}80.53	^{4}79.00	^{5}81.89	^{4}78.16	^{5}74.55	^{5}56.72	^{2}83	^{4}29.80
b	Raptor (SEM+MO, FW)	^{2}98.96	^{2}84.97	^{2}84.54	^{2}88.51	^{2}86.12	^{2}89.29	^{3}84.49	^{2}84.97	^{2}81.83	^{4}53.64	^{4}103	^{3}38.60

DAVIS-Moving

exp.	method	bg	obj	P	R	F	Pu	Ru	Fu	AO	FN	FP	AP
a	Rigid mask (mono) [46]	^{6}92.68	^{7}50.78	^{7}66.59	^{7}54.63	^{7}54.55	^{7}59.98	^{7}50.50	^{7}50.78	^{7}47.68	^{7}50.41	^{7}3199	^{7}4.20
a	Towards [14]	^{1}96.40	^{1}79.37	^{3}81.71	^{2}85.34	^{2}82.31	^{1}78.65	^{2}82.65	^{1}79.37	^{2}73.97	^{3}18.89	^{1}549	^{4}33.00
a	Towards w/o tracking [14]	^{2}94.90	^{5}72.57	^{5}79.96	^{4}80.31	^{4}78.35	^{5}69.11	^{3}82.12	^{5}72.57	^{5}70.11	^{2}18.25	^{6}3075	^{5}20.80
a+b	**Raptor**	^{4}94.87	^{3}75.93	^{2}82.74	^{3}82.11	^{3}80.84	^{2}75.90	^{4}79.67	^{3}75.93	^{3}73.16	^{4}20.99	^{2}1638	^{2}40.80
b	Raptor (SEM)	^{5}94.25	^{2}76.13	^{1}84.25	^{1}87.86	^{1}85.00	^{4}72.73	^{1}85.38	^{2}76.13	^{1}77.59	^{1}14.80	^{5}2979	^{1}45.40
b	Raptor (MO, FW)	^{7}92.48	^{6}55.34	^{6}68.93	^{6}58.73	^{6}58.72	^{6}64.78	^{6}54.39	^{6}55.34	^{6}52.63	^{6}49.22	^{3}1852	^{6}14.00
b	Raptor (SEM+MO, FW)	^{3}94.88	^{4}74.17	^{4}80.45	^{5}78.49	^{5}77.80	^{3}74.73	^{5}77.38	^{4}74.17	^{4}70.49	^{5}23.62	^{4}2339	^{3}37.30

YTVOS-Moving

exp.	method	bg	obj	P	R	F	Pu	Ru	Fu	AO	FN	FP	AP
a	Rigid mask (mono) [46]	^{7}83.62	^{7}18.70	^{7}32.77	^{7}18.21	^{7}19.71	^{7}29.88	^{7}17.48	^{7}18.70	^{7}34.42	^{7}79.94	^{6}2096	^{7}1.60
a	Towards [14]	^{1}92.06	^{2}74.41	^{3}81.27	^{2}77.55	^{2}77.98	^{2}77.74	^{3}74.97	^{2}74.41	^{3}75.89	^{3}22.11	^{2}543	^{3}33.50
a	Towards w/o tracking [14]	^{5}88.83	^{3}69.24	^{2}82.32	^{3}76.36	^{3}77.05	^{3}68.73	^{2}76.83	^{3}69.24	^{5}70.54	^{2}17.98	^{7}2990	^{5}20.20
a+b	**Raptor**	^{3}90.47	^{4}60.35	^{4}67.51	^{4}63.50	^{4}64.10	^{4}64.43	^{4}60.94	^{4}60.35	^{2}76.20	^{4}36.37	^{4}760	^{2}40.00
b	Raptor (SEM)	^{2}91.33	^{1}78.95	^{1}86.19	^{1}83.70	^{1}83.45	^{1}79.21	^{1}83.56	^{1}78.95	^{1}78.58	^{1}12.37	^{5}1363	^{1}57.00
b	Raptor (MO, FW)	^{6}85.02	^{6}22.46	^{6}33.49	^{6}21.39	^{6}23.58	^{6}32.74	^{6}20.05	^{6}22.46	^{6}45.74	^{6}78.55	^{3}752	^{6}4.90
b	Raptor (SEM+MO, FW)	^{4}89.28	^{5}48.00	^{5}53.09	^{5}50.20	^{5}50.55	^{5}51.33	^{5}48.08	^{5}48.00	^{4}75.35	^{5}50.16	^{1}415	^{4}31.90

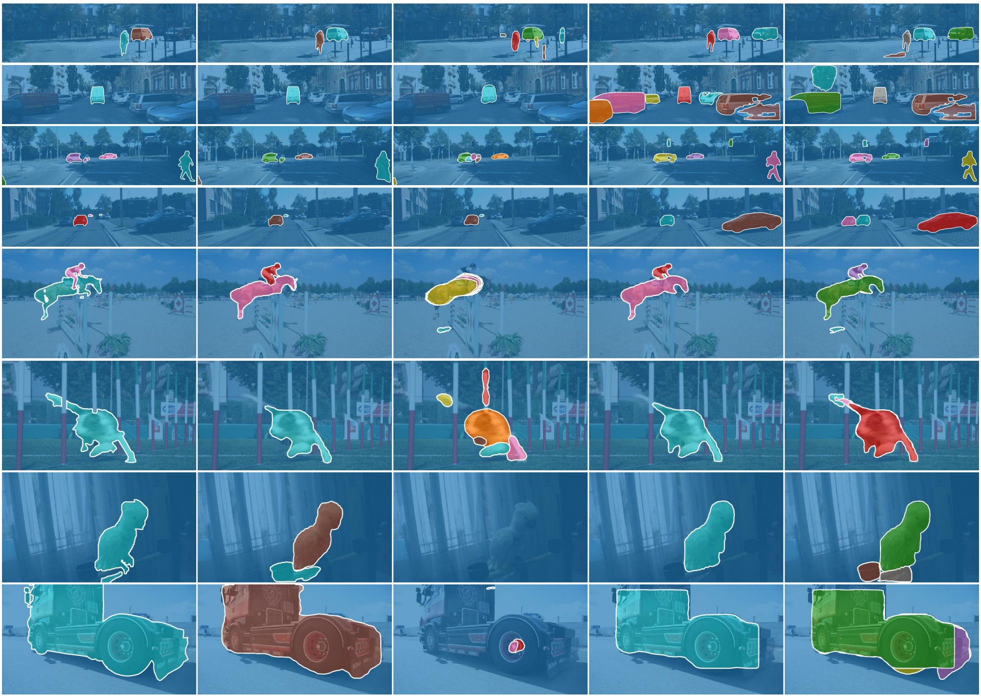
Table 1: Raptor performance (**exp.** a) and ablation study (**exp.** b) on several datasets. Results are coloured from best (green) to worst (red). See the description of the metrics in Section 4.3. The left upper index shows the rank of the method. (Best viewed in colour.)

We also adopt background (bg) intersection over union (IoU) and objects IoU (obj) metrics used by [46]. For obj, first the best match between the ground truth and predicted segmentation is found, then IoU is computed for all objects and averaged. To see better the segmentation precision of successfully detected objects (IoU with GT greater than 50), we introduce the *average overlap* (AO) measure which computes average IoU over these detections only.

The above measures focus on the segmentation quality, so we present also classical *false positive* (FP) and *false negative* (FN) measures to see the number of falsely detected or missed GT objects. We further compute the *average precision* (AP) adopted from the COCO evaluation [21] as a single number summary of the previous two metrics.

4.4 Results

For comparison we choose the two best performing methods for the instance motion segmentation task. Both state-of-the-art methods, "Towards" [14] and "RigidMask" (mono) [46], which we compare to, have shown results outperforming other prior work. As they did not originally evaluate on the same datasets and using the same metrics, our evaluation also shows their relative strengths and weaknesses on top of comparing them to Raptor. Towards [14] uses offline tracking of the whole sequence, both forward and backward in time, as the final stage of the estimation process. For a fair comparison with Raptor and other state-



GT | **Raptor** | RigidMask [46] | Towards [14] | Towards [14], w/o tracking

Figure 3: Sample results on KITTI (rows 1-4), DAVIS-Moving (rows 5,6) and YTVOS-Moving (rows 7,8) datasets. RigidMask fails to detect non-rigidly moving objects and produces high number of FPs even on parts of rigid scenes. Towards [14] fails in forward moving camera scenes, where it often detects static semantically meaningful objects. Without tracking, Towards outputs FPs on all datasets. Raptor does not suffer from these problems.

of-the-art methods we evaluate also a versions of Towards without tracking (*w/o tracking* in the tables).

We show results on individual datasets in Table 1. Figure 3 presents examples segmentations for Raptor and the compared methods. The KITTI dataset tests a forward camera motion through a structured environment, DAVIS-Moving contains a variety of object classes and the camera is usually following the main subject, while YTVOS-Moving is composed of rather difficult sequences with partially visible objects and background which is sometimes barely visible and often difficult for the geometrical approaches.

Clearly, Raptor works well on all metrics across all datasets. In particular, it excels in AP and AO measures which shows its ability to discover objects (AP) and its ability to segment well the correctly (IoU > 50) found objects (AO).

Towards is the best on both DAVIS-Moving and YTVOS-Moving, but it used the training parts of these datasets for training and uses the offline tracking. Without tracking, the ranking swaps on DAVIS-Moving and the gap between Raptor and Towards reduces on YTVOS-Moving. The tables also show how much is Towards relying on the tracking for FP reduction. Towards is significantly worse on both KITTI datasets, demonstrating its inability to cope with complex camera motion and to generalise beyond the training datasets.

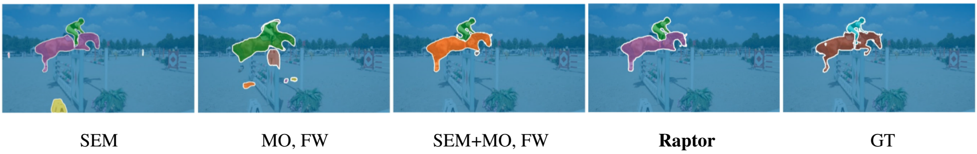


Figure 4: Ablation of Raptor component. We refer to Section 4.4 for details.

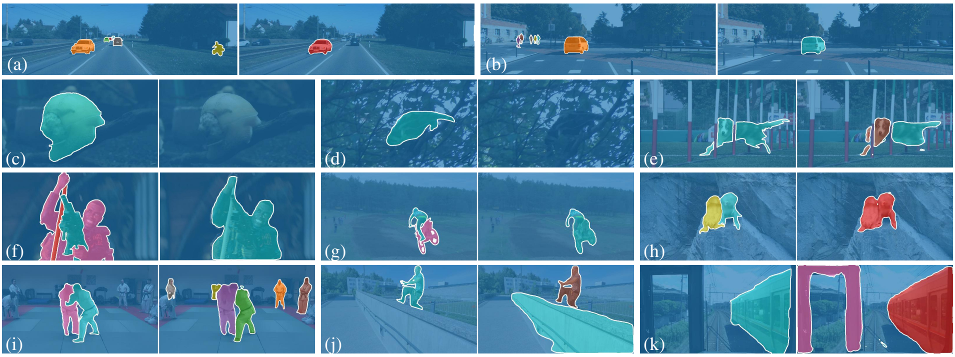


Figure 5: Example of Raptor failure cases. The left of the image pairs shows GT and the right the estimated segmentation. The most common failure cases are due to a small apparent motion (a,b,c). Others problems include: obstacles (d,e), merging moving object instances (f,g,h), false-positive detection for semantically meaningful objects (i) and optical flow failure (j). Sometimes, the GT is incomplete and we discover missing objects (k).

The RigidMask performs reasonably well on the original KITTI annotations containing only rigidly moving objects, but the performance degrades on the extended KITTI where pedestrians and buses are annotated as well. It fails on the other two datasets, where the high FP values indicate that it over-segments the scene into rigid parts (see Figure 3).

The commonly used pixel-based measures (bg, obj, P, R, F, Pu, Ru, and Fu) are influenced not only by the good detections as AO or AP, but also the "quality" of FPs and the number of FNs. A slightly weaker values of these measures on DAVIS and YTVOS compared to Towards indicate the problem of the motion branch to generate reasonable MCV when the scene is covered almost completely by the object or the background is not static (water, trees, ...). Being trained on the FlyingThings3D dataset only, the network has never seen such scenes. However, the success of Towards on such sequences indicates that adding similar data to training may improve results in such cases.

Interestingly, the class-agnostic instance semantic detector Raptor (SEM) is top ranked in almost all metrics on the YTVOS-moving dataset. This indicates that only a minority of semantically meaningful objects in the dataset are marked as “non-moving”, which we consider a weakness of the dataset not noticed before. We also found that many objects marked as “moving” do not move between some frames of the sequence, which results in a higher number of FNs for all methods.

Raptor failure cases. (see Figure 5) The most common problem are slowly moving objects, objects far from the camera or objects close to the point-of-expansion. Their detection would probably require longer temporal integration. Another problem observed is merging of ambiguous object instances (a person with a bag, motorcycle and its driver, etc.). Raptor also struggles with significantly occluded objects (behind leafs, bars, ...) and sometimes returns static but semantically meaningful objects.

Ablation study. Table 1 shows in its bottom parts the results of the ablation study. To demonstrate the effectiveness of the combination of the semantic and motion branches, we tested also a motion branch alone (MO, FW). It works reasonably well for the KITTI dataset, similarly to the RigidMask method, but fails on non-rigid motion datasets like DAVIS and YTVOS as expected. The tables also show that the backward MCV improves performance compared to the forward version only (SEM+MO, FW) across the datasets. Typical results of individual configurations are shown in Figure 4.

Time complexity. Currently, Raptor (including all individual modules) runs 3.1s[3] per KITTI-size image on average and needs about 5GB VRAM. The training took 12 days for 12 epochs for class-agnostic branch[4]. Pre-training of the motion backbone on the MODaS problem took 2 days with 6 epochs and final Raptor stage training took 3 days with 6 epochs.

5 Conclusions

We proposed Raptor, a novel CNN architecture for the MODaS problem. It has both semantic and geometrical components. That allow Raptor outputs semantically meaningful moving object segmentation from a monocular video. Raptor shows a strong generalisation ability across a range of environments and datasets. It achieves state-of-the-art results on standard benchmarks using an extensive set of measures. We show that it benefits from using not only forward, but also backward motion information. The network discovers both rigid and non-rigid objects. We see this result as a first step towards monocular scene understanding. For instance, in the autonomous driving scenario it is not enough to discover moving objects, but it is also necessary to estimate their 3D trajectories. Given a monocular video, this is the next challenge to be considered.

6 Acknowledgement

The research was supported by CTU student grant SGS20/171/OHK3/3T/13 and by Toyota Motor Europe HS.

References

[1] A. Ayvaci and S. Soatto. Motion segmentation with occlusions on the superpixel graph. In *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*, pages 727–734, Sept 2009. doi: 10.1109/ICCVW.2009.5457630.

[2] Alper Ayvaci and Stefano Soatto. Detachable object detection with efficient model selection. In *Energy Minimization Methods in Computer Vision and Pattern Recognition*, pages 191–204. Springer, 2011.

[3] Intel(R) Core(TM) i7-8700K CPU @ 3.70GHz with a single NVIDIA GTX 1080Ti graphic card

[4] Intel(R) Core(TM) i7-8700K CPU @ 3.70GHz with 2x NVIDIA GTX 1080Ti graphic cards

[3] A.G. Bors and I. Pitas. Prediction and tracking of moving objects in image sequences. *Image Processing, IEEE Transactions on*, 9(8):1441 –1445, aug 2000. ISSN 1057-7149. doi: 10.1109/83.855440.

[4] G.D. Borshukov, G. Bozdagi, Y. Altunbasak, and A.M. Tekalp. Motion segmentation by multistage affine classification. *Image Processing, IEEE Transactions on*, 6(11): 1591 –1594, nov 1997. ISSN 1057-7149. doi: 10.1109/83.641420.

[5] T. E. Boult and L. Gottesfeld Brown. Factorization-based segmentation of motions. In *Proceedings of the IEEE Workshop on Visual Motion*, pages 179–186, 1991. doi: 10.1109/WVM.1991.212809.

[6] Eric Brachmann and Carsten Rother. Neural-guided ransac: Learning where to sample model hypotheses. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4322–4331, 2019.

[7] Thomas Brox and Jitendra Malik. Object segmentation by long term analysis of point trajectories. In *European conference on computer vision*, pages 282–295. Springer, 2010.

[8] Thomas Brox, Andrés Bruhn, and Joachim Weickert. Variational motion segmentation with level sets. In *Computer Vision–ECCV 2006*, pages 471–483. Springer, 2006.

[9] A. Bugeau and P. Perez. Detection and segmentation of moving objects in highly dynamic scenes. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition CVPR '07*, pages 1–8, 2007. doi: 10.1109/CVPR.2007.383244.

[10] Aurélie Bugeau and Patrick Pérez. Track and cut: simultaneous tracking and segmentation of multiple objects with graph cuts. *J. Image Video Process.*, 2008:3:1–3:14, January 2008. ISSN 1687-5176. doi: 10.1155/2008/317278. URL http://dx.doi.org/10.1155/2008/317278.

[11] Kai Chen, Jiangmiao Pang, Jiaqi Wang, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jianping Shi, Wanli Ouyang, et al. Hybrid task cascade for instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4974–4983, 2019.

[12] João Paulo Costeira and Takeo Kanade. A multibody factorization method for independently moving objects. *International Journal of Computer Vision*, 29:159–179, September 1998. ISSN 0920-5691. doi: 10.1023/A:1008000628999. URL http://dl.acm.org/citation.cfm?id=299660.299666.

[13] Trevor Darrell and Alexander Pentland. Robust estimation of a multi-layered motion representation. In *Proceedings of the IEEE Workshop on Visual Motion*, pages 173–174. IEEE Computer Society, 1991.

[14] Achal Dave, Pavel Tokmakov, and Deva Ramanan. Towards segmenting anything that moves. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[15] Benjamin Drayer and Thomas Brox. Object detection, tracking, and motion segmentation for object-level video segmentation. *arXiv preprint arXiv:1608.03066*, 2016.

[16] K. Fragkiadaki and Jianbo Shi. Detection free tracking: Exploiting motion and topology for segmenting and tracking under entanglement. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, pages 2073–2080, 2011. doi: 10.1109/CVPR.2011.5995366.

[17] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3354–3361, June 2012. doi: 10.1109/CVPR.2012.6248074.

[18] Abdul Mueed Hafiz and Ghulam Mohiuddin Bhat. A survey on instance segmentation: state of the art. *International Journal of Multimedia Information Retrieval*, pages 1–19, 2020.

[19] Michal Irani, Benny Rousso, and Shmuel Peleg. Detecting and tracking multiple moving objects using temporal integration. In *European Conference on Computer Vision*, pages 282–287. Springer, 1992.

[20] Suyog Dutt Jain, Bo Xiong, and Kristen Grauman. Fusionseg: Learning to combine motion and appearance for fully automatic segmention of generic objects in videos. *arXiv preprint arXiv:1701.05384*, 2017.

[21] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[22] Nikolaus Mayer, Eddy Ilg, Philip Hausser, Philipp Fischer, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *CVPR*, 2016.

[23] Moritz Menze and Andreas Geiger. Object scene flow for autonomous vehicles. In *CVPR*, 2015.

[24] Michal Neoral, Jan Šochman, and Jiří Matas. Continual occlusions and optical flow estimation. *arXiv preprint arXiv:1811.01602*, 2018.

[25] P. Ochs and T. Brox. Higher order motion models and spectral clustering. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 614 –621, june 2012. doi: 10.1109/CVPR.2012.6247728.

[26] P. Ochs, J. Malik, and T. Brox. Segmentation of moving objects by long term video analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(6): 1187–1200, June 2014. ISSN 0162-8828. doi: 10.1109/TPAMI.2013.242.

[27] Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 724–732, 2016.

[28] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alex Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017.

[29] Siyuan Qiao, Liang-Chieh Chen, and Alan Yuille. DetectoRS: Detecting objects with recursive feature pyramid and switchable atrous convolution. *arXiv preprint arXiv:2006.02334*, 2020.

[30] Jianbo Shi and J. Malik. Motion segmentation and tracking using normalized cuts. In *Computer Vision, 1998. Sixth International Conference on*, pages 1154 –1160, jan 1998. doi: 10.1109/ICCV.1998.710861.

[31] Mennatullah Siam, Heba Mahgoub, Mohamed Zahran, Senthil Yogamani, Martin Jagersand, and Ahmad El-Sallab. Modnet: Moving object detection network with motion and appearance for autonomous driving. *arXiv preprint arXiv:1709.04821*, 2017.

[32] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. *arXiv preprint arXiv:2003.12039*, 2020.

[33] Pavel Tokmakov, Karteek Alahari, and Cordelia Schmid. Learning motion patterns in videos. *arXiv preprint arXiv:1612.07217*, 2016.

[34] Philip HS Torr. Geometric motion segmentation and model selection. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 356(1740):1321–1340, 1998.

[35] R. Tron and R. Vidal. A benchmark for the comparison of 3-d motion segmentation algorithms. In *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pages 1 –8, june 2007. doi: 10.1109/CVPR.2007.382974.

[36] Yi-Hsuan Tsai, Ming-Hsuan Yang, and Michael J Black. Video segmentation via object flow. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3899–3908, 2016.

[37] R. Vidal and R. Hartley. Motion segmentation with missing data using powerfactorization and gpca. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pages II–310 – II–316 Vol.2, june-2 july 2004. doi: 10.1109/CVPR.2004.1315180.

[38] Sudheendra Vijayanarasimhan, Susanna Ricco, Cordelia Schmid, Rahul Sukthankar, and Katerina Fragkiadaki. Sfm-net: Learning of structure and motion from video. *arXiv preprint arXiv:1704.07804*, 2017.

[39] Zhexiong Wan, Yuxin Mao, and Yuchao Dai. Praflow_rvc: Pyramid recurrent all-pairs field transforms for optical flow estimation in robust vision challenge 2020. *arXiv preprint arXiv:2009.06360*, 2020.

[40] J.Y.A. Wang and E.H. Adelson. Representing moving images with layers. *Image Processing, IEEE Transactions on*, 3(5):625–638, Sep 1994. ISSN 1057-7149. doi: 10.1109/83.334981.

[41] Christopher Xie, Yu Xiang, Zaid Harchaoui, and Dieter Fox. Object discovery in videos as foreground motion clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9994–10003, 2019.

[42] Ning Xu, Linjie Yang, Yuchen Fan, Jianchao Yang, Dingcheng Yue, Yuchen Liang, Brian Price, Scott Cohen, and Thomas Huang. Youtube-vos: Sequence-to-sequence video object segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 585–601, 2018.

[43] Xun Xu, Loong Fah Cheong, and Zhuwen Li. Motion segmentation by exploiting complementary geometric models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2859–2867, 2018.

[44] Jingyu Yan and Marc Pollefeys. A general framework for motion segmentation: Independent, articulated, rigid, non-rigid, degenerate and non-degenerate. In *Computer Vision - ECCV 2006*, volume 3954 of *Lecture Notes in Computer Science*, pages 94–106. Springer Berlin / Heidelberg, 2006. ISBN 978-3-540-33838-3. URL http://dx.doi.org/10.1007/11744085_8.

[45] Gengshan Yang and Deva Ramanan. Volumetric correspondence networks for optical flow. In *Advances in Neural Information Processing Systems*, pages 793–803, 2019.

[46] Gengshan Yang and Deva Ramanan. Learning to segment rigid motions from two frames. *arXiv preprint arXiv:2101.03694*, 2021.

[47] Zhenheng Yang, Peng Wang, Yang Wang, Wei Xu, and Ram Nevatia. Every pixel counts: Unsupervised geometry learning with holistic 3d motion understanding. *arXiv preprint arXiv:1806.10556*, 2018.