MOFT: Monocular odometry based on deep depth and careful feature selection and tracking

Karlo Koledić, Igor Cvišić, Ivan Marković, and Ivan Petrović[1].

Abstract— **Autonomous localization in unknown environments is a fundamental problem in many emerging fields and the monocular visual approach offers many advantages, due to being a rich source of information and avoiding comparatively more complicated setups and multisensor calibration. Deep learning opened new venues for monocular odometry yielding not only end-to-end approaches but also hybrid methods combining the well studied geometry with specific deep components. In this paper we propose a monocular odometry that leverages deep depth within a feature based geometrical framework yielding a lightweight frame-to-frame approach with metrically scaled trajectories and state-of-the-art accuracy. The front-end is based on a multihypothesis matcher with perspective correction coupled with deep depth predictions that enables careful feature selection and tracking; especially of ground plane features that are suitable for translation estimation. The back-end is based on point-to-epipolar line minimization for rotation and unit translation estimation, followed by deep depth aided reprojection error minimization for metrically correct translation estimation. Furthermore, we also present a domain shift adaptation approach that allows for generalization over different camera intrinsic and extrinsic setups. The proposed approach is evaluated on the KITTI and KITTI-360 datasets, showing competitive results and in most cases outperforming other state-of-the-art stereo and monocular methods.**

I. INTRODUCTION

Localization of an autonomous agent within an unknown environment is a fundamental problem in various emerging fields, such as autonomous driving and virtual reality. Most methods use a combination of proprioceptive and exteroceptive sensors. While fusion of information from many complementary sensors offers increased robustness and accuracy compared to a single sensor, it also comparatively complicates sensor setup and calibration. Cameras offer rich information about immediate surrounding, while being relatively cheap and widely available, thus motivating localization from a single sensor within Visual Odometry (VO) or Visual Simultaneous Localization and Mapping (V-SLAM) frameworks.

Most VO methods belong to the class of feature-based approaches, which extract a sparse set of image features and match them across multiple frames [1]–[3]. On the other hand, direct methods [4], [5] optimize photometric error, therefore skipping costly pre-processing step and achieving superior performance in textureless regions. The classic geometric VO frameworks are often challenged by scenarios that include motion blur, textureless regions, occlusions, and many dynamic objects. Given that, with advancements in deep learning, learning-based methods have been proposed to tackle these issues. By leveraging large amount of data, neural networks can extract appropriate features, thus enabling robustness even in difficult scenarios. End-to-end learning has been used in both supervised [6] and self-supervised formulations [7], [8], where networks usually jointly learn depth maps and ego-motion. However, end-to-end deep learning based solutions generally underperform compared to geometric methods. The most obvious issue is that it is challenging to enforce convolutional networks to learn the well studied and formulated geometric background of the problem and the networks will usually overfit due to the biased data distribution in the training set [9]. Although providing networks with sufficiently diverse data may mitigate the overfitting problem [10], the collection of data with sufficiently diverse camera intrinsics and extrinsincs in distinctive environments is by and large impractical.

In order to circumvent these issues, works such as [11]–[14] use deep learning, particularly depth prediction networks, within well-formulated geometric frameworks in order to complement its shortcomings and increase robustness. This is particularly useful in monocular VO, where predicted depth significantly reduces the often encountered scale-drift. Additionally, by inducing metric scale into the training process (e.g., by using stereo sequences [15]), networks can produce accurately scaled depth maps during inference with monocular inputs. This enables such hybrid systems to produce correctly scaled trajectories while using only a single camera. For example, D3VO [11] achieves such formulation via tight integration of Monodepth2 [15] inspired deep network and DSO [5].

In order to estimate metrically scaled trajectories, depth prediction networks are frequently trained with stereo sequences, making the training self-supervised and collection of data within distinctive automotive environments relatively straightforward. However, hybrid methods do not consider possible variations in camera extrinsics and intrinsics compared to the training setup. In [16]–[18] it was shown that depth prediction quality reduces with different cameras, even in similar environments. In order to estimate the metrically scaled trajectory, the monocular camera used during inference would need to have exact parameters as its stereo counterpart used in training.

[1] Authors are with University of Zagreb Faculty of Electrical Engineering and Computing, Laboratory for Autonomous Systems and Mobile Robotics, Zagreb, Croatia {name.surname@fer.hr}.

This work has been supported by the European Regional Development Fund under the grant KK.01.2.1.02.0119 – Research and development of an advanced unit for autonomous control of mobile vehicles in logistics (A-UNIT) and KK.01.1.1.01.0009 - Advanced methods and technologies in Data Science and Cooperative Systems (DATACROSS)

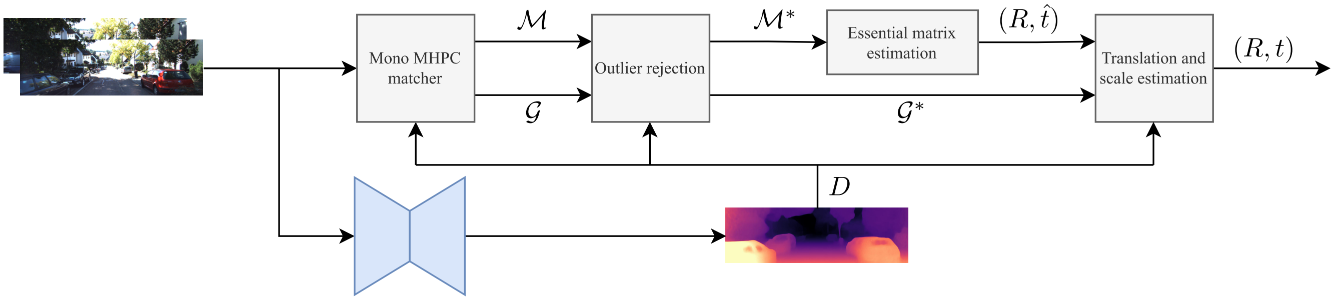


Fig. 1: Proposed monocular visual odometry system.

Various attempts have been made to improve generalization of depth networks during training by embedding camera parameters within neural networks [17]–[19]. Unfortunately, such approaches would still require a high amount of hardly obtainable diverse data. On the other hand, [20], [21] enforce depth map consistency with known camera height. In [20] authors estimate ground plane and formulate a dense geometrical constraint to help recover the scale during test time. However, they focus their method for scale recovery of networks trained with monocular sequences, not considering domain shift introduced by usage of different camera during test time. In [21] online retraining is used to align estimated and ground truth camera heights. This should in theory adequately adjust network weights after domain shift, but requires computationally expensive and complicated training process for each new sensor combination. Note that it is possible to estimate metrically correct trajectory, while only requiring knowledge of camera height to infer scale [22]; however, such methods add additional computational complexity of homography estimation and assume local planarity of the ground plane, which can lead to unreliable estimation due to sidewalks and other objects in urban environments. Deep networks on the other hand can address such anomalies.

In this paper, we propose a feature based monocular odometry with deep depth predictions (MOFT) related to our stereo solution SOFT2 [23] – currently the highest ranking odometry on the KITTI dataset. We use deep depth prediction to improve monocular robustness and enable estimation of metrically scaled trajectories. Unlike previous works [11], [13], [14], we develop the method to be robust with regards to perturbations in camera intrinsics and extrinsics. The main contributions of the proposed approach are as follows:

- a monocular hybrid VO system with modified multihypothesis matcher with perspective correction [23] leveraging deep depth for fast and accurate matching, requiring only two images to estimate a correctly scaled relative camera pose
- a domain shift adaptation algorithm based on default camera height, allowing our method to generalize well for different vehicle–camera setups
- evaluation on the KITTI and KITTI-360 datasets showing competitive results and in most cases outperforming other state-of-the-art stereo and monocular methods.

II. Proposed hybrid monocular odometry

In this section we describe our proposed monocular VO system and our approach for domain shift adaptation. We assert that single-image depth prediction can enhance monocular feature-based VO on three levels: (i) feature matching, (ii) outlier rejection, and (iii) scale aware translation optimization. We modify our MHPC matcher proposed in [23] for monocular sequences, while using depth predictions for faster and more accurate matching. Compared to other state-of-the-art approaches, our matcher naturally detects plenty of ground plane features, which enables accurate and robust translation optimization without scale drift that plagues many monocular methods. In [4], [24] authors use loop-closure and bundle adjustment to reduce accumulated scale-drift, while our method functions like a pure odometry, requiring only two images to estimate the relative camera pose (we do not use any kind of multi-frame optimization). Our proposed pipeline is depicted in Fig. 1

Afterwards, we propose a domain shift adaptation algorithm for our monocular hybrid VO, allowing our method to generalize well for different car/camera setups. In particular, we consider most frequent changes of camera parameters in the context of road vehicles: camera height, camera pitch, focal length and principal point. We notice that, for points in the close vicinity of the camera, such domain shift causes a constant bias in the neural network depth prediction on the ground plane. This bias can be calculated via the known camera height and used to refine the depth prediction before its usage in front-end tracking and back-end optimization. Finally, similarly to other hybrid methods [11], in this work we use Monodepth2 [15] inspired architecture for depth prediction. We train our network in the self-supervised manner with the stereo sequences, thus allowing metrically correct translation optimization.

A. MOFT front-end

Sparse feature-based methods rely heavily on accurate feature matching across adjacent frames. To have robust rotation and translation estimation a distinctive set of features is required. Distant features exhibit motion in the image plane only during rotational movement and influence significantly estimation of the rotational part of the motion. In order to accurately resolve the translational part of the motion, features that are relatively close to the vehicle should be matched with high accuracy since these features have higher parallax during translational motion.

Additionally, it would be beneficial for the system to know which features reside on the ground plane. Depth uncertainty is commonly highest on non-Lambertian surfaces and moving objects [11], which do not exist on the ground plane. Furthermore, we argue that the deep network exhibits best generalization ability exactly for ground plane parts of the image, since other image regions can contain objects and scenes unseen in the training set, which often lead to spurious depth predictions.

The multiple hypotheses perspective correction (MHCP) matcher, presented in [23], was developed for our stereo visual odomety SOFT2 and relied on stereo image pairs. In this paper we present the monocular version that relies on a pair of sequential images and the depth predicted from the deep network. In the following we describe the monocular MHCP matcher, while the pseudocode is given in Algorithm 1. Firstly, we detect distinctive features with standard Shi-Thomasi corner detector [25]. We select 2 strongest corners for each 50×50 pixels large bin in the image. With depth map D predicted by the deep network, we calculate surface normals for each feature. Given that, by searching for features with normals that are perpendicular to the camera motion, we can establish possible candidates for ground-plane futures.

Features detected in current frame with coordinates (u, v), as well as points in their neighborhoods, i.e., points $\{(u-1, v-1), (u, v-1), ..., (u+1, v+1)\}$, are projected to 3D space with the following equation:

$$P(u,v) = \begin{bmatrix} (u - c_x) \cdot \frac{D(u,v)}{f_x} \\ (v - c_y) \cdot \frac{D(u,v)}{f_y} \\ D(u,v) \end{bmatrix}, \tag{1}$$

where $P(u, v)$ refers to the 3D coordinates of the detected feature (u, v), with $(f_x, f_y), (c_x, c_y)$ being corresponding camera focal length and principal point parameters. Similarly to [20], for each detected feature we create a set of normals from vectors that form a 90-degree angle when projected to image coordinates, $S_i = \{(P(u+1, v) - P(u, v)) \times (P(u, v+1) - P(u, v)), ...\}$, where i represents index of the feature. Finally, we average and normalize to estimate the surface normal n_i:

$$n_i = \frac{\sum_j n_j / ||n_j||_2}{|S_i|}, \tag{2}$$

where n_j is a j-th element of set S_i. We use 8 points from the immediate neighborhood, which gives 4 combinations of orthogonal vectors in the image plane. Ground points in automotive scenarios are generally orthogonal to the camera motion and we classify the i'th feature as a ground point candidate if $|\alpha_i| > \alpha_{min}$, where

$$\alpha_i = \arccos(n_i \cdot \check{t}) \tag{3}$$

and $\check{t}$ refers to normalized camera translation vector between the current and previous frame. As this is unknown, we initialize it with the constant velocity model, which is sufficiently accurate in automotive localization. After we establish whether feature is a possible ground point, we proceed with our feature matching strategy.

Algorithm 1 Mono MHPC matcher

```
Require: Images $\mathcal{I}$, $\mathcal{I}^-$; odometry from the previous step $(\check{R}, \check{t})$; predicted depths $D$.
Ensure: Matched features $\mathcal{M}$, ground plane matches $\mathcal{G}$

Detect strong and evenly distributed features:
$\mathcal{F} \leftarrow \mathrm{get\_features}(\mathcal{I})$
for $i = 1 : |\mathcal{F}|$ do
    calculate $n_i$ by Eq.(2)
    if $|\arccos(n_i \cdot \check{t})| > \alpha_{min}$ then
        Features get two hypothetical patch transforms:
        $\mathcal{F}_i' \leftarrow \{\mathrm{gnd\_tf}(\mathcal{F}_i, \check{R}, \check{t}, D), \mathrm{norm\_tf}(\mathcal{F}_i, \check{R}, \check{t}, D)\}$
    else
        $\mathcal{F}_i' \leftarrow \mathrm{norm\_tf}(\mathcal{F}_i, \check{R}, \check{t}, D)$
    end if
    Compute NCC near the projection:
    $(\mathcal{F}^-, \mathrm{ncc}) \leftarrow \mathrm{local\_ncc}(\mathcal{I}^-, \mathcal{F}_i', \check{R}, \check{t}, D)$
    $(\mathrm{ncc}) \leftarrow \mathrm{sort}(\mathrm{ncc})$
    $\mathrm{diff\_scores} \leftarrow \mathrm{diff}(\mathrm{ncc})$
    $k \leftarrow \mathrm{find\_index}(\mathrm{diff\_scores} > \mathrm{diff\_th})$
    if $k > 10 \; || \; k = \emptyset$ then
        $\mathcal{M}_i \leftarrow \emptyset$
    else
        Select the match with the highest score:
        $\mathcal{M}_i \leftarrow \{\mathcal{F}_i, \mathcal{F}^-(\&max(ncc))\}$
        if $\mathrm{ground\_hypothesis}(\mathcal{M}_i)$ then
            $\mathcal{G}_i = \mathcal{M}_i$
        end if
    end if
end for
```

For each feature classified as a candidate ground point, we generate patch predictions based on two hypotheses: feature is either on the ground plane or not. When generating predictions for the ground plane hypothesis we assume orthogonality of the patch normal and camera motion vector. Otherwise, we assume that the patch resides on a plane with the normal vector pointing towards the camera. Note that we generate patch predictions for both hypotheses only if the feature is classified as a candidate ground point, i.e, we do not generate patch predictions for the ground plane hypothesis if features failed the test given by (3). This greatly reduces the complexity of the original MHPC matcher.

Patch predictions are correlated in positions within a narrow envelope around projection of the predicted point in the previous image via normalized cross correlation (NCC). If there is no significant difference between the highest 10 scores, we classify feature as ambiguous and discard it. Otherwise, we select the match with the best overall score. In case of features within the candidate ground point set, we classify them as ground points $\mathcal{G}$ if the higher NCC score comes from the ground plane hypothesis. Thus, to be classified as a ground point, feature needs to pass both the depth and photometric consistency checks.

To summarize, we use predicted depth in multiple stages in order to enhance the matcher accuracy, robustness, and decrease computational complexity. Our original MHPC matcher computes ground plane 3D points via the intersection of the feature back-projected ray with the ground plane, which requires expensive homography estimation. Predicted depth allows us to compute 3D points in a simple manner as in (1). On top of that, depth information enables significant reduction of the area where detected feature may reside in the previous image. Other monocular methods either track features across multiple frames or search along the entire epipolar line, which increases the cost notably and creates spurious matches. Finally, depth in the immediate neighborhood of the feature is used as an additional constraint in the classification of the ground plane features. Decision based on photometric constraint alone sometimes leads to false classifications [23]. By inclusion of depth information, we classify ground futures more reliably leading to better performance in the back-end optimization.

B. MOFT back-end

With the set of correspondences $\{(x_i, x_i')\}_{i=1}^N$ established during front-end tracking, we seek to optimize the rotation and metrically scaled translation of the camera. However, first we want to detect features with spurious matches and inconsistent depth predictions. For this purpose, we use the difference in 3D space to quantify feature uncertainty

$$U(x_i) = ||\pi^{-1}(x_i', D(x_i')) - \check{R}\pi^{-1}(x_i, D(x_i)) + \check{t}||_2^2, \quad (4)$$

where π represents projection operator, with π^{-1} functioning as in (1) and $(\check{R}, \check{t})$ initialized with a constant velocity model. Features that do not satisfy a threshold are classified as outliers and removed from set of active features, obtaining $\mathcal{M}^*$ and $\mathcal{G}^*$. Note that inclusion of depths $D(x_i')$ and $D(x_i)$ creates a constraint which enforces consistency between the assumed camera motion and depths predicted with the deep network. This, in addition to detection of false matches, identifies features on moving vehicles that, even when correct, negatively impact the estimation process. Removal of such matches increases robustness and greatly decreases the computational complexity during RANSAC iterations in the subsequent optimization.

With having outliers removed, we proceed with our SOFT2 approach for rotation estimation, where we iteratively estimate the essential matrix within a RANSAC framework. The essential matrix is parameterized in the following way

$$E(\xi) = E(\alpha, \beta, \gamma, \theta, \phi) = R(\alpha, \beta, \gamma)[\hat{t}(\theta, \phi)]_\times, \quad (5)$$

where (α, β, γ) are the Euler angles and (θ, ϕ) are spherical coordinates of the translation vector. We formulate our optimization function as minimization of point-to-epipolar-line distances and the objective function is written as follows

$$\min_{R,\hat{t}} \sum_i d^2\left(x_i, l_i'(\xi)\right) + d^2\left(x_i', l_i(\xi)\right), \quad (6)$$

where $d(x, l)$ represents the point-to-epipolar-line distance, $l_i'(\xi) = E(\xi)^\top x_i'$ and $l_i(\xi) = E(\xi)x_i$ are epipolar lines associated to points x_i' and x_i in the previous and current view, respectively. We use all feature matches which survive the outlier rejection procedure. The objective function is nonlinear and thus optimized with the Levenberg-Marquardt algorithm within a RANSAC framework. Note that translation parameters are also optimized, but only up to scale.

Afterwards, we seek to optimize metrically correct translation parameters. Optimization is formulated as minimization of the reprojection error

$$e(x_i, x_i') = x_i - \pi(R\pi^{-1}(x_i', D(x_i')) + t), \quad (7)$$

where rotation matrix R is already calculated in the previous step within the essential matrix estimation, while t is initialized with an up to scale translation vector $\hat{t}$ estimated in the same step. We minimize the symmetric squared reprojection error for all ground plane features that survived the outlier rejection

$$\min_t \sum_i \sigma_i^{-2}(e(x_i, x_i')^T e(x_i, x_i') + e(x_i', x_i)^T e(x_i', x_i)), \quad (8)$$

with σ_i^2 being covariance proportional to the feature depths. Inclusion of the predicted depth D within optimization allows us to finally estimate metrically scaled translation vector. We use only active ground plane features for translation optimization due to, as previously mentioned, low uncertainty of the depth prediction on the ground plane.

C. Camera height alignment

Deep learning enhanced VO and V-SLAM algorithms can generalize well across different environments, due to large amount of data available for self-supervised training. However, generalization across different camera models has proven to be extremely difficult [18]. In this paper we propose an approach for alignment of estimated and ground truth camera height, which enables our algorithm to achieve state-of-the-art results even with different camera parameters than on the training sequences.

As mentioned earlier, depth prediction generalizes well for the ground plane with respect to the environmental factors. In comparison, here we focus on changes in the camera parameters for automotive scenarios that would affect predicted depth map, i.e., the camera height, focal length, vertical principal point position and camera pitch. We assert that a shift in the stated intrinsic and extrinsic parameters causes a constant bias in the ground plane depth prediction. **We show that for nearby ground points, this can be parameterized as a constant scale factor, without significant degradation to the VO performance**. Note that we assume that these changes are within reasonable boundaries, i.e., they do not heavily distort the scene appearance.

In order to estimate this scale factor, we choose a sequence of N frames with clearly visible ground plane without significant perturbations. For each frame we project all points to 3D space using (1), and then we calculate normals and classify points as ground points as in (3). Unlike [20], which uses median of heights calculated from a set of ground points, we seek to find the dominant ground plane.

TABLE I: Results for the KITTI Odometry sequences (M – monocular, S – stereo, and D – using depth prediction networks)

		01		02		06		08		09		10	
		t_{rel}	r_{rel}	t_{rel}	r_{rel}	t_{rel}	r_{rel}	t_{rel}	r_{rel}	t_{rel}	r_{rel}	t_{rel}	r_{rel}
DSO [5]	M	9.17	-	114	-	42.2	-	188	-	28.1	-	24.0	-
ORB-SLAM [1]	M	107.57	0.89	10.34	0.26	14.56	0.26	11.46	0.28	9.30	0.26	2.57	0.32
S-DSO [26]	S	1.43	**0.09**	0.78	**0.21**	0.67	**0.20**	**0.98**	**0.25**	0.98	0.18	**0.49**	**0.18**
S-LSD-VO [27]	S	2.13	0.37	1.09	0.37	1.28	0.43	1.24	0.38	1.22	0.28	0.75	0.34
ORB-SLAM2 [24]	S	1.44	0.19	0.77	0.28	0.89	0.27	1.03	0.31	0.86	0.25	0.62	0.29
OV2-SLAM [28]	S	3.70	0.29	0.79	0.22	1.13	0.28	1.11	0.31	0.96	0.20	0.52	0.18
DF-VO [29]	M + D	56.76	13.93	2.38	0.39	1.03	0.30	1.60	0.32	2.61	0.29	2.29	0.37
DVSO [14]	M + D	1.18	0.11	0.84	0.22	0.71	0.20	1.03	0.25	0.83	0.21	0.74	0.21
D3VO [11]	M + D	1.07	-	0.80	-	**0.67**	-	1.00	-	0.78	-	0.62	-
MOFT (Ours)	M + D	**0.90**	0.16	**0.74**	0.24	0.73	0.25	1.05	0.27	**0.69**	**0.18**	0.84	0.26

This allows us to filter out points which would generate inaccurate height estimations due to violating the ideal plane assumption.

Given that, we minimize point-to-plane distances

$$\min_{n_j, h_j} \sum_i \frac{|n_j^T P_i + h_j|}{||n_j||} \tag{9}$$

within a RANSAC framework, where n_j refers to the ground plane normal at j'th frame. We estimate the plane with a 3 point hypothesis and classify points as inliers if the distance is within 0.01 m. Each point is given a score which is proportional to the inverse depth. Finally, we choose the set of inliers with the highest cumulative score. This encourages the inclusion of nearby points when choosing inliers for the dominant ground plane calculation.

In the end, we estimate the camera height h_j for all inliers using (9) and the scale factor is then calculated as

$$\lambda_j = h^* / h_j, \tag{10}$$

where h^* represents known camera height. We repeat this process for every frame in the sequence, obtaining $\{\lambda_j\}_{j=1}^N$. Outliers are filtered out via median absolute deviation and the final parameter λ is calculated as the mean of the surviving inliers.

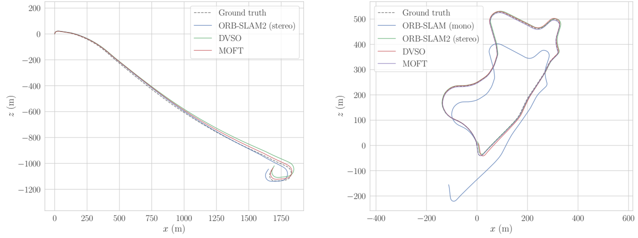
After obtaining the ground plane bias parameters, our odometry is ready to be run on sequences with different camera models compared to the training dataset. To achieve this, we adjust depths of the ground plane features during test time in the following manner

$$z' = \lambda z. \tag{11}$$

Furhtermore, robust feature matching also requires accurate depth predictions for those features that are not on the ground plane. We noticed that these depths are largely unaffected by camera height change and camera rotation, thus we adjust the depths of these features with a simple focal length normalization

$$z' = \frac{f_y}{f_y^*} z, \tag{12}$$

where f_y^* represents focal length of the camera used in the training sequences.



(a) KITTI 01 sequence

(b) KITTI 09 sequence

Fig. 2: Estimated trajectories on KITTI Odometry sequences

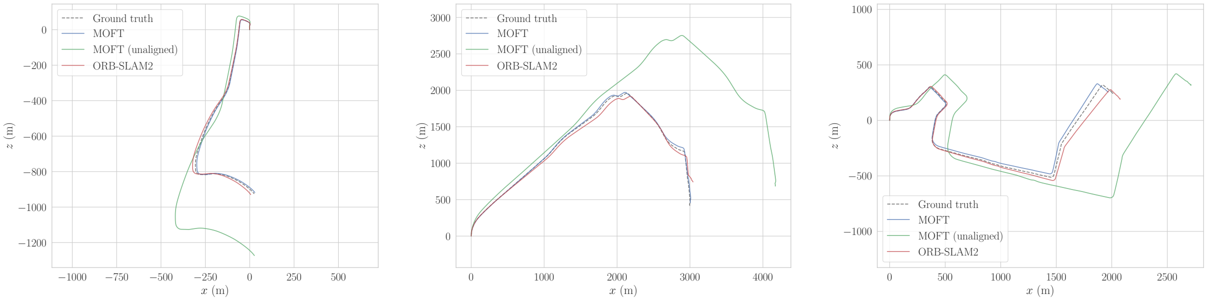
III. Experimental Evaluation

We evaluated MOFT on the KITTI [30] and KITTI-360 [31] datasets. First, we compared our method on the KITTI sequences to state-of-the-art monocular, stereo and hybrid methods methods. Afterwards, we tested our generalization ability for different camera setups on the KITTI-360 dataset and compared it to the stereo ORB-SLAM2 as it has a different camera setup in both intrinsic and extrinsic camera parameters (with respect to the ground plane).

A. KITTI dataset

We trained the self-supervised depth prediction on the Eigen split [32] and used Monodepth2 [15] architecture to achieve fair comparison with other hybrid methods. Our model was trained with stereo sequences in order to estimate the metric scale. The network was trained in PyTorch [33], with inference implemented in its C++ API to decrease execution time and enable integration with the rest of our system. We ran the deep network in parallel with front-end tracking and back-end optimization, thus allowing for real-time execution. We tested the method on sequences 01, 02, 06, 08, 09 and 10 of the KITTI Odometry benchmark, as they are not contained within the Eigen split. As proposed in [30], we used relative translational (t_{rel}) and rotational error (r_{rel}) for evaluation.

In Table I we show results for the KITTI sequences. For open source implementations of [1], [24], [28] we used the default parameters and turned off loop closure in the SLAM methods [1], [24], [28] in order to focus on odometry



(a) KITTI-360 03 sequence

(b) KITTI-360 07 sequence

(c) KITTI-360 10 sequence

Fig. 3: Estimated trajectories of the KITTI-360 Odometry sequences.

TABLE II: Results for the KITTI-360 Odometry sequences

	MOFT		ORB-SLAM2	
	t_{rel}	r_{rel}	t_{rel}	r_{rel}
00	0.453	0.156	**0.333**	**0.149**
02	**0.548**	**0.211**	0.584	0.226
03	0.682	**0.149**	**0.486**	0.169
04	0.552	0.220	**0.515**	**0.216**
05	0.567	0.250	**0.459**	**0.247**
06	**0.510**	**0.167**	0.523	0.177
07	**0.682**	**0.141**	5.075	0.973
09	**0.774**	**0.180**	1.073	0.184
10	**1.457**	**0.234**	1.730	0.434

and achieve fair comparison. Both direct [5] and feature-based [1] monocular methods showed high degree of scale drift, which is reflected in the translational error. This, in addition to the unknown absolute scale, makes these methods dependent on fusion with another sensor. Monocular ORB-SLAM [1] failed on the highway sequence 01 and showed high scale drift on sequence 09, as can be seen in Fig. 2. In comparison, deep depth prediction enabled our method to estimate accurately the metrically scaled translation without scale-drift, while being a purely monocular method at test time. **MOFT generally outperformed stereo and deep learning enhanced methods in the translational error, while having comparatively weaker rotational error results. This is most likely due to the current lack of multi-frame optimizations which are often present in other approaches. In spite of that, as a result of our matching strategy involving monocular depth estimation, we are able to achieve superior translational error results**.

We are also the first hybrid method that achieves comparable and in some cases better results than D3VO and DVSO. We do not present results for other hybrid methods, as they achieve lesser accuracy, but a summary can be found in [34].

B. KITTI-360

Here we focus on the generalization ability of our method with respect to camera intrinsic and extrinsic parameters, which is a frequent scenario in autonomous driving. We selected the KITTI-360 dataset since it has similar environment compared to KITTI, on which we trained the depth prediction network, but uses a camera with significantly different focal length and principal point parameters, while being mounted on a car at different height and with a slight downward camera inclination of 5 degrees. Even though the scene context is not dramatically changed, the change of the camera parameters has a considerable effect on the predicted depth maps. **Here, we show that our camera height alignment procedure enables accurate metrically-scaled estimation under such circumstances**.

In order to perform the camera height alignment, we chose a segment of the sequence 03 with a wide and highly planar road scene to calculate the scale factor λ. We tested our method and compared it to stereo ORB-SLAM2. **We did not compare the proposed approach with other hybrid methods, e.g., [11], [14], since they do not report results that include variations of camera parameters during test time.** Results are presented in Table II, where we can see that the results are on par with ORB-SLAM2 and more than half sequences better. Furthermore, in Fig. 3 we show the estimated trajectories, before and after camera height alignment. The alignment allowed us to estimate scaled trajectories correctly even in the presence of the domain shift due to the different camera parameters.

IV. Conclusion

In this paper we have presented a feature based monocular odometry with deep depth predictions that is related to our stereo odometry SOFT2 [23] – currently the highest ranking odometry on the KITTI dataset. It is based on feature tracking using monocular multihypothesis matcher with perspective correction coupled with deep depth that enables selection of quality ground plane features particularly suitable for translation estimation. Using point-to-epipolar-line error minimization we estimate first rotation and translation direction, which is followed then by deep depth aided reprojection error minimization for estimating the metrically correct trajectory. Additionally, we proposed a domain shift adaptation for changes in the camera intrinsic and extrinsic parameters, enabling the method to work when faced with different vehicle–camera setup in test time. Results on the KITTI and KITTI-360 datasets validated the approach and showed competitive results and in majority of cases exceeded state-of-the-art monocular and stereo approaches.

References

[1] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "Orb-slam: a versatile and accurate monocular slam system," *IEEE transactions on robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[2] I. Cvišić and I. Petrović, "Stereo odometry based on careful feature selection and tracking," in *2015 European Conference on Mobile Robots (ECMR)*. IEEE, 2015, pp. 1–6.

[3] G. Klein and D. Murray, "Parallel tracking and mapping for small ar workspaces," in *2007 6th IEEE and ACM international symposium on mixed and augmented reality*. IEEE, 2007, pp. 225–234.

[4] J. Engel, T. Schöps, and D. Cremers, "Lsd-slam: Large-scale direct monocular slam," in *European conference on computer vision*. Springer, 2014, pp. 834–849.

[5] J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 3, pp. 611–625, 2017.

[6] S. Wang, R. Clark, H. Wen, and N. Trigoni, "Deepvo: Towards end-to-end visual odometry with deep recurrent convolutional neural networks," in *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2017, pp. 2043–2050.

[7] R. Li, S. Wang, Z. Long, and D. Gu, "Undeepvo: Monocular visual odometry through unsupervised deep learning," in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 7286–7291.

[8] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, "Unsupervised learning of depth and ego-motion from video," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1851–1858.

[9] S. Li, X. Wang, Y. Cao, F. Xue, Z. Yan, and H. Zha, "Self-supervised deep visual odometry with online adaptation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 6339–6348.

[10] W. Wang, Y. Hu, and S. Scherer, "Tartanvo: A generalizable learning-based vo," *arXiv preprint arXiv:2011.00359*, 2020.

[11] N. Yang, L. v. Stumberg, R. Wang, and D. Cremers, "D3vo: Deep depth, deep pose and deep uncertainty for monocular visual odometry," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1281–1292.

[12] K. Tateno, F. Tombari, I. Laina, and N. Navab, "Cnn-slam: Real-time dense monocular slam with learned depth prediction," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 6243–6252.

[13] S. Y. Loo, A. J. Amiri, S. Mashohor, S. H. Tang, and H. Zhang, "Cnn-svo: Improving the mapping in semi-direct visual odometry using single-image depth prediction," in *2019 International conference on robotics and automation (ICRA)*. IEEE, 2019, pp. 5218–5223.

[14] N. Yang, R. Wang, J. Stuckler, and D. Cremers, "Deep virtual stereo odometry: Leveraging deep depth prediction for monocular direct sparse odometry," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 817–833.

[15] C. Godard, O. Mac Aodha, M. Firman, and G. J. Brostow, "Digging into self-supervised monocular depth estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 3828–3838.

[16] T. v. Dijk and G. d. Croon, "How do neural networks see depth in single images?" in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 2183–2191.

[17] Y. Zhao, S. Kong, and C. Fowlkes, "Camera pose matters: Improving depth prediction by mitigating pose distribution bias," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 15 759–15 768.

[18] J. M. Facil, B. Ummenhofer, H. Zhou, L. Montesano, T. Brox, and J. Civera, "Cam-convs: Camera-aware multi-scale convolutions for single-view depth," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 11 826–11 835.

[19] L. He, G. Wang, and Z. Hu, "Learning depth from single images with deep neural network embedding focal length," *IEEE Transactions on Image Processing*, vol. 27, no. 9, pp. 4676–4689, 2018.

[20] F. Xue, G. Zhuo, Z. Huang, W. Fu, Z. Wu, and M. H. Ang, "Toward hierarchical self-supervised monocular absolute depth estimation for autonomous driving applications," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 2330–2337.

[21] B. Wagstaff and J. Kelly, "Self-supervised scale recovery for monocular depth and egomotion estimation," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 2620–2627.

[22] R. Tian, Y. Zhang, D. Zhu, S. Liang, S. Coleman, and D. Kerr, "Accurate and robust scale recovery for monocular visual odometry based on plane geometry," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 5296–5302.

[23] I. Cvišić, I. Marković, and I. Petrović, "Soft2: Stereo visual odometry for road vehicles based on a point-to-epipolar-line metric," *IEEE Transactions on Robotics*, 2022.

[24] R. Mur-Artal and J. D. Tardós, "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras," *IEEE transactions on robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

[25] J. Shi *et al.*, "Good features to track," in *1994 Proceedings of IEEE conference on computer vision and pattern recognition*. IEEE, 1994, pp. 593–600.

[26] R. Wang, M. Schworer, and D. Cremers, "Stereo dso: Large-scale direct sparse visual odometry with stereo cameras," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 3903–3911.

[27] J. Engel, J. Stückler, and D. Cremers, "Large-scale direct slam with stereo cameras," in *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2015, pp. 1935–1942.

[28] M. Ferrera, A. Eudes, J. Moras, M. Sanfourche, and G. Lebesnerais, "OV^{2}SLAM : A Fully Online and Versatile Visual SLAM for Real-Time Applications," *IEEE Robotics and Automation Letters*, 2021.

[29] H. Zhan, C. S. Weerasekera, J.-W. Bian, and I. Reid, "Visual odometry revisited: What should be learnt?" in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 4203–4210.

[30] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.

[31] Y. Liao, J. Xie, and A. Geiger, "Kitti-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[32] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," *Advances in neural information processing systems*, vol. 27, 2014.

[33] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," 2017.

[34] C. Chen, B. Wang, C. X. Lu, N. Trigoni, and A. Markham, "A survey on deep learning for localization and mapping: Towards the age of spatial machine intelligence," *arXiv preprint arXiv:2006.12567*, 2020.