

000
001
002
003
004
005
006
007
008
009
010
011
012
013
014
015
016
017
018
019
020
021
022
023
024
025
026
027
028
029
030
031
032
033
034
035
036
037
038
039
040
041
042
043
044
045
046
047
048
049
050
051
052
053054
055
056
057
058
059
060
061
062
063
064
065
066
067
068
069
070
071
072
073
074
075
076
077
078
079
080
081
082
083
084
085
086
087
088
089
090
091
092
093
094
095
096
097
098
099
100
101
102
103
104
105
106
107

On the Consistency of Self-Supervised Depth and Motion Estimation

Anonymous CVPR submission

Paper ID 1366

Abstract

The self-supervised learning of depth and pose from monocular sequence provides an attractive solution by using the photometric consistency of nearby frames as it depends much less on the ground-truth data. In this paper, we show that the photometric consistency is often not sufficiently competent to guarantee a good result because of the dynamic nature of real world scenarios. Then, we introduce several new loss terms from indirect self-generated geometric supervision to multi-view consistency inferred from the networks. As demonstrated on commonly used benchmarks, the proposed method achieves by far the best performance among the methods that simultaneously estimate monocular depth and relative poses.

1. Introduction

Along with the great success of deep neural networks on various semantic vision tasks such as recognition and segmentation [21], recent deep methods have interestingly shown promising abilities in modeling geometric relations, such as encoding depth and pose information. These geometric tasks used to have enough good closed-form solutions, yet some learning-based approaches are shown to improve traditional methods [51, 5] and have the potential to solve more challenging geometric tasks such as monocular depth estimation [42].

The joint learning of depth and relative pose from monocular videos [47, 52, 55] has been a heated research area due to its key role in *simultaneous localization and mapping* (SLAM) and *visual odometry* (VO) applications. The simplicity and the unsupervised nature make itself a potential replacement for traditional approaches that involve complicated geometric computations. Given adjacent frames, this approach uses convolutional neural networks (CNNs) to jointly predict the depth map of the target image and the relative poses from the target image to its visible neighboring frames. With the target image depth and relative poses, it minimizes the photometric error between the original target image and the synthesized images formed by

bilinear-sampling [22] the adjacent views.

However, several existing problems hinder the performance of this type of methods. First, the photometric loss formulation requires the modeling scene to be static without non-Lambertian surfaces or occlusions. This assumption is often violated in imagery without careful photometric calibration, as well as street-view datasets [7, 15] with moving cars and pedestrians. Second, as the depth inference considers only a single image, there is no guarantee that adjacent frames would result in consistent depth estimation. Third, even though the method works on N -view images, only pairwise information is utilized in the training process. As a result, the scale of estimated $(N - 1)$ relative poses is solely determined by the estimated depth of target image.

In this paper, we address the geometric consistency of self-supervised depth and pose estimation, seeking to mitigate or resolve the above limitations. We are mostly interested in the *unsupervised* monocular setting because it does not require stereo pairs or the costly ground-truth depth data. It is sometimes also referred to as the *self-supervised* approach since the supervisory signal inherits from the raw pixel or self-generated intermediate computations in traditional SfM methods [28]. We will use the two terms ‘self-supervised’ and ‘unsupervised’ interchangeably throughout the paper. Apart from the self-supervised photometric information that is used as the main source of supervision in the learning, we show that the intermediate geometric computation such as feature matches can also be employed in the training pipeline and greatly improve the performance. Our method is inspired by the several lines of research works. The general architecture is built upon the recent self-supervised learning approaches [55] that make use of photometric loss for depth and pose estimation. Yet, the motion module is inspired by indirect local-feature-based relative pose estimation and epipolar geometry [18, 20, 39]. The proposed formulation emphasizes the geometric consistency of deep interplay between pose and depth.

The consistency in multi-view data, or referred to as the data association problem [6], has been widely applied to and even forms the basis for many sub-steps in SfM, from feature matching [3], view graph construction [44, 53], mo-

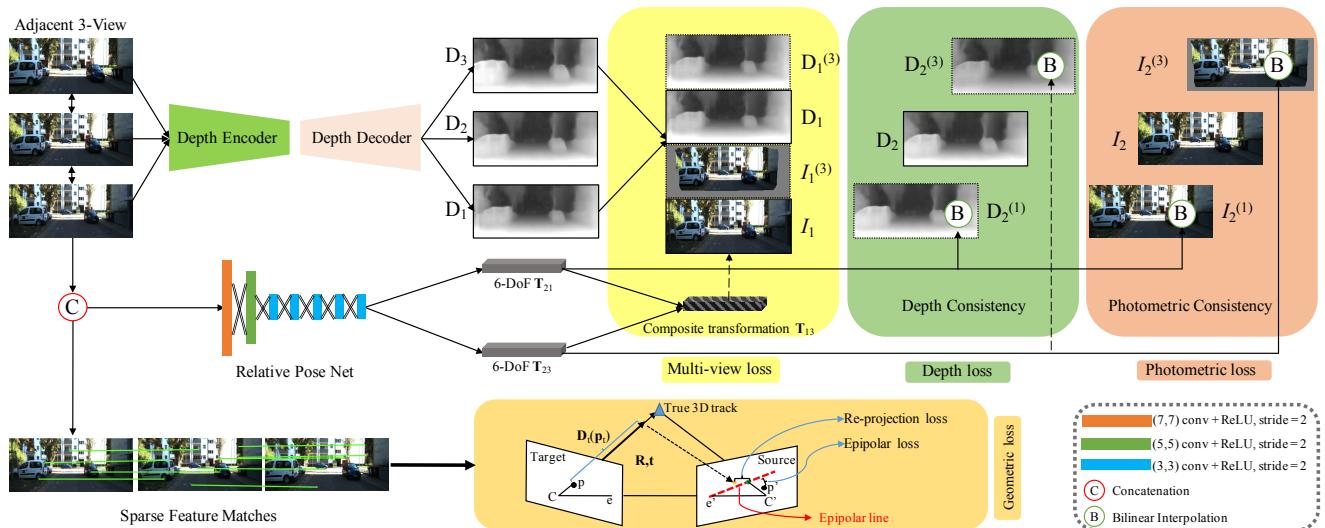


Figure 1. The architecture of our method. Our end-to-end training pipeline involves multiple loss terms to achieve photometric consistency, geometric consistency and multi-view consistency.

tion averaging [17] to bundle adjustment [45]. Yet enforcing the consistency is non-trivial in the learning-based setting. Our contributions are summarized twofold: 1) we incorporate sparse feature matches to the self-supervised learning framework, which introduces epipolar error and re-projection error to complement the noisy photometric loss; 2) we propose a novel depth estimation formulation that aims for consistent outputs of adjacent views. Throughout extensive evaluations on several benchmark datasets [7, 15], we show that the deep interaction of multi-view depths and poses achieves the state-of-the-art performance.

2. Related Works

Structure-from-Motion and Visual SLAM. Structure-from-Motion (SfM) [2] and visual SLAM problems aim to simultaneously recover the camera pose and 3D structures from images. Both problems are well studied and render practical systems [10, 37, 50] by different communities for decades, with the latter emphasizes more on the real-time performance. The self-supervised depth and motion learning framework derives from direct SLAM methods [10, 11, 38]. Different from indirect methods [8, 29, 37] that use reliable sparse intermediate geometric quantities like local features [41], direct methods optimize the geometry using all the pixels in the image. With accurate photometric calibration such as gamma and vignetting correction [26], this formulation does not rely on sparse geometric computation and is able to generate finer-grained geometry. However, this formulation is less robust than indirect ones when the photometric loss is not meaningful, the scene containing moving or non-Lambertian objects.

Supervised Approaches for Learning Depth. Some early monocular depth estimation works rely on information from depth sensors [9, 42] without the aid of geometric relations. Liu *et al.* [32] combine deep CNN and conditional random field for estimating single monocular images. DeMoN [46] is an iterative supervised approach to jointly estimate optical flow, depth and motion. This coarse-to-fine process considers the use of stereopsis and produces good results with both depth and motion supervision.

Unsupervised/Self-Supervised Approaches. The self-supervised approaches for SfM stem from warping-based view synthesis [56], a classical paradigm of which is to composite novel view based on the underlying 3D geometry. Garg *et al.* [14] propose to learn depth using stereo camera pairs with known relative pose, yet the depth learning formulation is unsupervised in nature. Godard *et al.* [16] improve this training paradigm with left-right consistency checking. The above two methods both rely on a calibrated stereo rig and are not applicable to monocular settings with unknown poses. The joint unsupervised optimization of depth and pose starts from Zhou *et al.* [55] and Vijayanarasimhan *et al.* [47]. They propose similar approaches that use two CNNs to estimate depth and pose separately, and constrain the outcome with photometric loss. Later, a series of improvements [28, 35, 48, 52, 54] are proposed. Wang *et al.* [48] discuss the scale ambiguity and combine the estimated depth with direct methods [10]. Zhan *et al.* [54] consider warping deep features from the neural nets instead of the raw pixel values. Klodt *et al.* [28] propose to integrate weak supervision from SfM methods. Mahjourian *et al.* [35] employ geometric constraints of the scene by en-

216 forcing an approximate ICP based matching loss. We shall
217 discuss and compare these unsupervised joint learning
218 approaches in detail in the experiments.
219

220 3. Method

221 3.1. Problem Formulation

222 We first formalize the problem and present effective
223 practices employed by previous methods [28, 35, 47, 48,
224 52, 54, 55]. Given adjacent N -view monocular image
225 sequences (e.g. $\{\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3\}$ for $N = 3$), the unsupervised
226 depth and motion estimation problem aims to simultaneously
227 estimate the depth \mathbf{D}_t of the target (center) image
228 (\mathcal{I}_2 in the 3-view case) and the 6-DoF relatives poses
229 $\mathbf{T}_{t \rightarrow s} = [\mathbf{R}_{t \rightarrow s} | \mathbf{t}_{t \rightarrow s}] \in \mathcal{SE}(3)$ to $N - 1$ source views (\mathcal{I}_1
230 and \mathcal{I}_3), using CNNs with photometric supervision.
231

232 For a source-target view pair $(\mathcal{I}_s, \mathcal{I}_t)$, \mathcal{I}_t can be inversely
233 warped to the source frame \mathcal{I}_s given the estimated depth \mathbf{D}_t
234 and the transformation from target to source $\mathbf{T}_{t \rightarrow s}$. Formally,
235 given a pixel coordinate p_t in \mathcal{I}_t which is co-visible
236 in \mathcal{I}_s , the pixel coordinate p_s in \mathcal{I}_s is given by the following
237 equation which determines the warping transformation
238

$$239 p_s \sim \mathbf{K}_s [\mathbf{R}_{t \rightarrow s} | \mathbf{t}_{t \rightarrow s}] \mathbf{D}_t(p_t) \mathbf{K}_t^{-1} p_t, \quad (1)$$

240 where \sim denotes ‘equal in the homogeneous coordinates’,
241 \mathbf{K}_s and \mathbf{K}_t are the intrinsics for the input image pair, and
242 $\mathbf{D}_t(p_t)$ is the depth for this particular p_t in \mathcal{I}_t .
243

244 With this coordinate transformation, synthesized images
245 can be generated from the source view using the differentiable
246 bilinear-sampling method [22]. The unsupervised
247 framework then minimizes the pixel error between the target
248 view and the synthesized image
249

$$250 \mathcal{L}_{pixel} = \frac{1}{|\mathcal{M}|} \sum_{\forall p_t \in \mathcal{M}} \left| \tilde{\mathcal{I}}_t^{(s)}(p_t | \mathbf{R}_{t \rightarrow s}, \mathbf{t}_{t \rightarrow s}, \mathbf{D}_t) - \mathcal{I}_t(p_t) \right|, \quad (2)$$

251 where $\tilde{\mathcal{I}}_t^{(s)}$ represents the synthesized target image from
252 source image. $\mathcal{I}(p)$ is the function that maps the image
253 coordinate p in image \mathcal{I} to pixel value, and the first term $\tilde{\mathcal{I}}_t^{(s)}$
254 is the bilinear-sampling operation used to acquire the synthesized
255 view given relative motion and depth. \mathcal{M} is a binary
256 mask that determines if the inverse warping fall into a valid
257 region in the source image, and can be computed analytically
258 given the per-pixel depth and relative transformation.
259 $|\mathcal{M}|$ denotes the total number of valid pixels.
260

261 In addition to the per-pixel error, structured similarity
262 (SSIM) [49] is shown to improve the performance [16, 52],
263 which is defined on local image patches x and y rather than
264 every single pixel. We follow the previous approaches [35,
265 52] to compute the SSIM loss on 3×3 image patches

($c_1 = 0.01^2, c_2 = 0.03^2$) as follows

$$266 \mathcal{L}_{SSIM}(\mathcal{I}_s, \mathcal{I}_t) = \frac{1}{2} \left(1 - \sum_{\forall x \in \mathcal{I}_s, \forall y \in \mathcal{I}_t} \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2\mu_y^2 + c_1)(\sigma_x + \sigma_y + c_2)} \right). \quad (3)$$

267 The depth map is further constrained by the smoothness
268 loss to push the gradients to propagate to nearby regions,
269 known as the gradient locality issue [4]. Specifically, we
270 adopt the image-aware smoothness formulation [52] which
271 allows sharper depth changes on edge regions
272

$$273 \mathcal{L}_{smooth} = \sum_{\forall p_t \in \mathcal{I}_t} |\nabla \mathbf{D}_t(p_t)|^T \cdot e^{-|\nabla \mathcal{I}_t(p_t)|}, \quad (4)$$

274 where ∇ denotes the 2D differential operator for computing
275 image gradients. Optimizing a combination of above loss
276 terms wraps up the basic formulation of training objectives,
277 which forms the baseline written as
278

$$279 \mathcal{L}_{baseline} = \alpha \mathcal{L}_{pixel} + (1 - \alpha) \mathcal{L}_{SSIM} + \beta \mathcal{L}_{smooth}. \quad (5)$$

280 However, there are drawbacks with the basic formulation.
281 We then describe the key ingredients our contributions.
282

283 3.2. Injecting Weak Geometric Supervision

284 The success of unsupervised joint training requires several
285 important assumptions: 1) The modeling scene should
286 be static without moving objects; 2) The surfaces in the
287 scene should be Lambertian; 3) No occlusion exists between
288 the target view and source view; 4) Cameras should
289 be photometrically calibrated, a technique adopted in the direct
290 SLAM method [10, 11], to compensate for vignetting
291 effect and exposure time. Violation to any of the above criteria
292 would lead to model bias or data bias. The first three
293 assumptions are inevitably violated to some extent because
294 it is hard to capture temporally static images with no occlusion
295 in the real world. The fourth restriction is often
296 neglected by datasets with no photometric calibration
297 parameters provided.
298

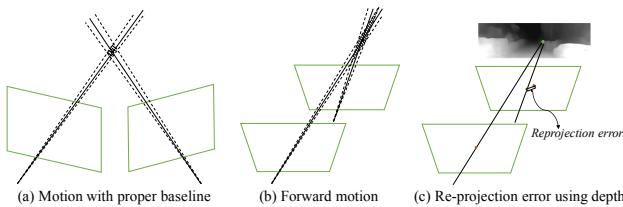
299 To address these limitations, previous methods [55, 28]
300 additionally train a mask indicating whether the photometric
301 loss is meaningful. Yet, we present a novel approach to
302 tackle this issue by injecting indirect geometric information
303 into the direct learning framework. Different from direct
304 methods that rely on dense photometric consistency, indirect
305 methods for SfM and visual SLAM are based on sparse
306 local descriptors such as SIFT [50] and ORB [37]. Local
307 invariant features are much less likely to be affected by the
308 scale and illumination changes and can be implicitly
309 embedded into the learning framework.
310

311 **Epipolar error.** Assuming the pinhole camera model, the
312 sparse feature matches $\mathcal{S}_{t \leftrightarrow s} = \{\mathbf{p} \leftrightarrow \mathbf{p}'\}$ between the
313 target and source views satisfy the epipolar constraint. Here
314 we assume the calibrated setting where pixel coordinates
315

324 are transformed to image coordinate by \mathbf{K}^{-1} . The loss with
 325 the feature matches and the estimated pose can be quantified
 326 using the *symmetric epipolar distance* [18]
 327

$$\mathcal{L}_{epi}(\mathcal{S}|\mathbf{R}, \mathbf{t}) = \sum_{\forall (\mathbf{p}, \mathbf{p}') \in \mathcal{S}} \left(\frac{\mathbf{p}'^T \mathbf{E} \mathbf{p}}{\sqrt{(\mathbf{E} \mathbf{p})_{(1)}^2 + (\mathbf{E} \mathbf{p})_{(2)}^2}} + \right. \\ \left. \frac{\mathbf{p}^T \mathbf{E} \mathbf{p}'}{\sqrt{(\mathbf{E} \mathbf{p}')_{(1)}^2 + (\mathbf{E} \mathbf{p}')_{(2)}^2}} \right), \quad (6)$$

334 where \mathbf{E} being the essential matrix computed by $\mathbf{E} = [t]_x \times \mathbf{R}$, $[\cdot]_x$ is the matrix representation of the cross
 335 product with t . We simply omit the subindices for conciseness
 336 (\mathcal{S} for $\mathcal{S}_{t \leftrightarrow s}$, \mathbf{R} for $\mathbf{R}_{t \rightarrow s}$, \mathbf{t} for $\mathbf{t}_{t \rightarrow s}$).
 337



344 Figure 2. (a) For two images with proper motion baseline, the
 345 uncertainty (shaded region) is small. (b) For forward motion with
 346 narrow baseline, the uncertainty is large. (c) The re-projection error
 347 unites estimated depth and pose with sparse features, and does
 348 not involve triangulation uncertainty.
 349

351
 352
 353 **Re-projection error.** The epipolar constraint does not
 354 concern the depth in its formulation. To involve depth op-
 355 timization using the feature match supervision, there are
 356 generally two methods: 1) triangulate the correspondence
 357 $\mathbf{p} \leftrightarrow \mathbf{p}'$ using the optimal triangulation method [18] assum-
 358 ing the Gaussian noise model, to obtain the 3D track for
 359 depth supervision; 2) back-project 2D features in one im-
 360 age using the estimated depth to compute the 3D track, and
 361 re-project the 3D track to another image to compute the re-
 362 projection error. For street-view ego-motion scenes [7, 15],
 363 points are less precisely triangulated since rays are almost
 364 parallel in forward motion (see Figure 2 for illustration).
 365 Therefore, we take the second method to weakly supervise
 366 the depth estimation by computing the re-projection loss
 367

$$\mathcal{L}_{reproj}(\mathcal{S}|\mathbf{R}, \mathbf{t}, \mathbf{D}_t) = \sum_{\forall p \leftrightarrow p' \in \mathcal{S}} \left\| [\mathbf{R}|\mathbf{t}] \hat{\mathbf{D}}_t(\mathbf{p}) \mathbf{p} - \mathbf{p}' \right\|_2, \quad (7)$$

370 where $\hat{\mathbf{D}}_t(\mathbf{p})$ is the bilinear-sampling operation [22] in the
 371 target depth map as the feature coordinate p is not an integer.
 372 Minimizing re-projection error using feature matches
 373 can be viewed as creating sparse anchors between the weak
 374 geometric supervision and the estimated depth and pose. In
 375 contrast, Equation 6 does not involve the estimated depth.
 376

377 We use the pre-computed pairwise matches filtered by
 378 geometric verification [19], yet outliers may still exist if
 379

380 they lie close to the epipolar line. However, minimizing
 381 the epipolar error and re-projection error in the neural net-
 382 works of all matches helps find a good hypothesis for pose
 383 estimation. Our experiments show that this weak supervi-
 384 sory signal significantly improves the pose estimation and is
 385 superior to using other intermediate SfM supervisions such
 386 as poses and sparse depth.
 387

388 3.3. Consistent Depth Estimation

389 In this section, we describe the depth estimation mod-
 390 ule. Previous methods, whether it is operated on 3-view
 391 or 5-view, are pairwise approaches in essence because loss
 392 terms are computed pairwisely from the source frame and
 393 target frame. Even though the pose network outputs $N - 1$
 394 relative poses at once, it is unknown that if these relative
 395 pose are aligned to the same scale. We propose the motion-
 396 consistent depth estimation formulation to address this is-
 397 sue. Rather than minimizing the loss between the target
 398 frame and adjacent source frames, our proposed formula-
 399 tion also considers the depth and motion consistency be-
 400 tween adjacent frames themselves.
 401

402 **Forward-backward consistency.** Inspired by [16, 40]
 403 that uses left-right consistency on stereo images, we pro-
 404 pose the forward-backward consistency for monocular im-
 405 ages. For the learning setting with stereo images, the images
 406 are rectified in advance so the scale ambiguity issue is not
 407 considered. While for learning monocular depth, the esti-
 408 mated depth is determined up to scale, therefore the align-
 409 ment of depth scale is necessary before constraining the
 410 depth discrepancy.
 411

412 Our network architecture estimates the depth maps of the
 413 target image (\mathcal{I}_t), and the forward and backward images. In
 414 addition to bilinear-sampling from the source image pixel
 415 values, we also sample the estimated depth maps of for-
 416 ward and backward images (\mathcal{I}_s). This process generates
 417 two synthesized depth maps $\tilde{\mathbf{D}}_t^{(s)}$ that can be used to con-
 418 strain the estimation of the target image depth \mathbf{D}_t . Since the
 419 monocular depth estimation network is agnostic to the depth
 420 scale of adjacent images, we apply a mean alignment to the
 421 synthesized depth maps and the target depth map given the
 422 analytical mask \mathcal{M} in Equation 2, and further optimize the
 423 depth discrepancy
 424

$$\mathcal{L}_{depth} = \frac{1}{|\mathcal{M}|} \sum_{\forall p \in \mathcal{M}} \left| \frac{mean(\mathbf{D}_t \circ \mathcal{M})}{mean(\tilde{\mathbf{D}}_t^{(s)} \circ \mathcal{M})} \cdot \tilde{\mathbf{D}}_t^{(s)}(p) - \mathbf{D}_t(p) \right|, \quad (8)$$

426 where \circ means the element-wise multiplication with the bi-
 427 nary mask and the loss is averaged over all the valid pixel p
 428 in the mask \mathcal{M} .
 429

Multi-view consistency. The above losses are all defined on the single target image (e.g. smoothness loss) or among image pairs, even though the input is N -view ($N \geq 3$) image sequences. The pose network output $N - 1$ relative poses between the target and source images, but the $N - 1$ relative poses are only weakly connected by the monocular depth. To strengthen the scale consistency for triplet relation, we propose the multi-view consistency loss which penalizes inconsistency of the forward depth and backward depth using the target image as a bridge for scale alignment. Formally, given image sequence $(\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3)$ with \mathcal{I}_2 the target image, and corresponding pose and depth predictions $(\mathbf{T}_{2 \rightarrow 1}, \mathbf{T}_{2 \rightarrow 3})$ and $(\mathbf{D}_1, \mathbf{D}_2, \mathbf{D}_3)$, we again obtained the normalized depth map $\bar{\mathbf{D}}_1 = s_{12} \cdot \mathbf{D}_1$ where the scaling ratio $s_{12} = \frac{\text{mean}(\mathbf{D}_2 \circ \mathcal{M}_{12})}{\text{mean}(\bar{\mathbf{D}}_1 \circ \mathcal{M}_{12})}$ as used in Equation 8. The transformation from the backward image \mathcal{I}_1 to the forward image \mathcal{I}_3 is $\mathbf{T}_{1 \rightarrow 3} = \mathbf{T}_{2 \rightarrow 1}^{-1} \cdot \mathbf{T}_{2 \rightarrow 3}$. The multiview loss minimizes the depth consistency term and photometric consistency term as

$$\begin{aligned} \mathcal{L}_{multi} = & \alpha \mathcal{L}_{pixel}(\mathcal{I}_3 | \bar{\mathbf{D}}_1, \mathbf{T}_{1 \rightarrow 3}) + (1 - \alpha) \mathcal{L}_{SSIM}(\mathcal{I}_3 | \bar{\mathbf{D}}_1, \mathbf{T}_{1 \rightarrow 3}) \\ & + \frac{1}{|\mathcal{M}_{13}|} \sum_{\forall p \in \mathcal{M}_{13}} \left| \bar{\mathbf{D}}_1(p) - \bar{\mathbf{D}}_1^{(3)}(p) \right|, \end{aligned} \quad (9)$$

where $\bar{\mathbf{D}}_1^{(3)}$ is the synthesized normalized depth for $\bar{\mathbf{D}}_1$ given $\bar{\mathbf{D}}_3$ and $\mathbf{T}_{1 \rightarrow 3}$. The subindices 1 and 3 are interchangeable in the above Equation 9. \mathcal{L}_{multi} goes beyond the pairwise loss terms \mathcal{L}_{pixel} , \mathcal{L}_{SSIM} , \mathcal{L}_{epi} and \mathcal{L}_{depth} because it utilizes the chained relative pose and pushes the two sub-relative poses to be aligned on the same scale. This benefits the monocular SLAM because it facilitates the incremental localization by aligning multiple N -view outputs.

3.4. Modeling the Data Uncertainty

The intrinsic noises (e.g. moving objects, occlusions and non-Lambertian surfaces) in the data pose challenges for the effectiveness of the photometric loss formulation. The proposed indirect supervision is able to mitigate the problem to some extend but still cannot fully solve it. Zhou *et al.* [55] tackle this weakness by training a mask to filter out probable erroneous pixels. We follow this procedure and reiterate it with a more formal representation using the *aleatoric* uncertainty theory. *Aleatoric* uncertainty [25] captures noise inherent in the observations, in contrast to epistemic uncertainty [13] which models the inherent noise in the model parameters. Specifically, heteroscedastic aleatoric uncertainty assume that the observation noise can vary with the input data, which fits our scenario where dynamic objects appear occasionally in the data. We use the Laplace distribution $f(\tilde{\mathcal{I}} | \mathcal{I}, \sigma(p_t)) = \frac{1}{2\sigma(p_t)} \exp(-\frac{|\tilde{\mathcal{I}} - \mathcal{I}|}{\sigma(p_t)})$ to model the heteroscedastic aleatoric uncertainty, whose negative log likelihood corresponds to the $L1$ minimization in Equation 2&3

Algorithm 1 Consistent Depth-Motion Training Loss	486
Require: $\{\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3\}$: input 3-view image snippet with \mathcal{I}_2 ($t = 2$) as the target image; $\mathcal{S} = \{\mathbf{p} \leftrightarrow \mathbf{p}'\}$: geometrically verified pairwise matches (\mathcal{S}_{21} and \mathcal{S}_{23})	487
Ensure: multi-view consistent depth \mathbf{D}_2 and relative poses $\mathbf{T}_{21} = \{\mathbf{R}_{21}, \mathbf{t}_{21}\}$, $\mathbf{T}_{23} = \{\mathbf{R}_{23}, \mathbf{t}_{23}\}$	488
1: Obtain the estimated depth map $\mathbf{D}_1, \mathbf{D}_2, \mathbf{D}_3$ and 6DoF pose vectors (Euler angles θ and displacement t), transform pose vector to matrix representation \mathbf{T}_{21} and \mathbf{T}_{23} .	489
2: $\mathbf{D}_2 := \mathbf{D}_2 / \text{median}(\mathbf{D}_2)$. Normalize the target depth \mathbf{D}_2 using, a critical step [48] to fix the scale of the image triplet.	490
3: Compute the \mathcal{L}_{pixel} , \mathcal{L}_{SSIM} pairwisely between $(\mathcal{I}_1, \mathcal{I}_2)$ and $(\mathcal{I}_2, \mathcal{I}_3)$, and \mathcal{L}_{smooth} on \mathcal{I}_2 given estimated depth and relative poses.	491
4: Compute \mathcal{L}_{epi} and \mathcal{L}_{reproj} pairwisely between $(\mathcal{I}_1, \mathcal{I}_2)$ and $(\mathcal{I}_2, \mathcal{I}_3)$ given feature matches \mathcal{S} .	492
5: First align the scales of \mathbf{D}_1 and \mathbf{D}_3 using Equation 8 and compute \mathcal{L}_{depth} pairwisely.	493
6: First compute the chained motion and normalized depth map $\bar{\mathbf{D}}_1$ and $\bar{\mathbf{D}}_3$, then compute \mathcal{L}_{multi} which is the only loss term that involves all the estimated quantities.	494
7: Losses involving per-pixel depth map are weighted by the data-dependent variance term $\sigma(p_t)$ as in Equation 10 and the total loss is given by Equation 11.	495
	496
	497
	498
	499
	500
	501
	502
	503
	504
	505
	506
	507
	508
	509
	510
	511
	512
	513
	514
	515
	516
	517
	518
	519
	520
	521
	522
	523
	524
	525
	526
	527
	528
	529
	530
	531
	532
	533

and the variance $\sigma(p_t)$ varies with respect to every pixel p_t . With the data-dependent variance term, we can for example rewrite the pixel error (Equation 2) using negative log likelihood as

$$\begin{aligned} \mathcal{L}_{pixel}^* \propto & \frac{1}{|\mathcal{M}|} \sum_{\forall p_t \in \mathcal{M}} \left(\frac{1}{\sigma(p_t)} \left| \tilde{\mathcal{I}}_t^{(s)}(p_t) - \mathcal{I}_t(p_t) \right| + \log \sigma(p_t) \right) \\ = & \frac{1}{|\mathcal{M}|} \sum_{\forall p_t \in \mathcal{M}} \left(\exp(-s(p_t)) \left| \tilde{\mathcal{I}}_t^{(s)}(p_t) - \mathcal{I}_t(p_t) \right| + s(p_t) \right). \end{aligned} \quad (10)$$

In practice, we compute the log variance $s(p_t) = \log \sigma(p_t)$ as an additional regression split from the last convolutional layer of the depth estimation network. We train the network to predict $\log \sigma(p_t)$ because it is claimed to be more numerically stable [25]. Our final formulation takes into account the basic losses in Equation 5, the feature matching terms, as well as the consistency terms, written as

$$\begin{aligned} \mathcal{L}_{total} = & \alpha \mathcal{L}_{pixel} + (1 - \alpha) \mathcal{L}_{pixel} + \beta \mathcal{L}_{smooth} + \\ & \gamma_1 \mathcal{L}_{epi} + \gamma_2 \mathcal{L}_{reproj} + \mu_1 \mathcal{L}_{depth} + \mu_2 \mathcal{L}_{multi}. \end{aligned} \quad (11)$$

For loss terms concerning per-pixel depth map, we can employ the data uncertainty similar to the one in Equation 10. We compare the total loss with and without the aleatoric uncertainty mask in the experiments. The weighting for different losses are set empirically given hyperparameters in previous methods and our attempts ($\alpha =$

540 Table 1. Single-view depth estimation performance. The statistics for the compared methods are excerpted from corresponding papers,
 541 except that the results marked with ‘*updated*’ are captured from the websites. ‘K’ represents KITTI raw dataset (Eigen split) and CS
 542 represents cityscapes training dataset. The method [16] marked with \star are trained and tested on larger scale (256×512) images, whereas
 543 others use 128×416 images. The metrics marked by red means ‘the lower the better’ and the ones marked by green means ‘the higher the
 544 better’. The best results for each category are **bolded**.

Method	Supervision	Dataset	Cap (m)	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Eigen <i>et al.</i> [9] Fine	Depth	K	80	0.203	1.548	6.307	0.282	0.702	0.890	0.958
Liu <i>et al.</i> [32]	Depth	K	80	0.202	1.614	6.523	0.275	0.678	0.895	0.965
Godard <i>et al.</i> [16] \star	Stereo/Pose	K	80	0.148	1.344	5.927	0.247	0.803	0.922	0.964
Godard <i>et al.</i> [16] \star	Stereo/Pose	K + CS	80	0.114	0.898	4.935	0.206	0.861	0.949	0.976
Zhou <i>et al.</i> [55] updated	No	K	80	0.183	1.595	6.709	0.270	0.734	0.902	0.959
Zhou <i>et al.</i> [55] updated	No	K	-	0.185	2.170	6.999	0.271	0.734	0.901	0.959
Klodt <i>et al.</i> [28]	No	K	80	0.166	1.490	5.998	-	0.778	0.919	0.966
Mahjourian <i>et al.</i> [35]	No	K	80	0.163	1.24	6.22	0.25	0.762	0.916	0.968
Wang <i>et al.</i> [48]	No	K	80	0.151	1.257	5.583	0.228	0.810	0.936	0.974
Yin <i>et al.</i> [52]	No	K	80	0.155	1.296	5.857	0.233	0.793	0.931	0.973
Yin <i>et al.</i> [52]	No	K	-	0.156	1.470	6.197	0.235	0.793	0.931	0.972
Yin <i>et al.</i> [52] updated	No	K + CS	80	0.149	1.060	5.567	0.226	0.796	0.935	0.975
Ours	No	K	80	0.140	1.014	5.473	0.222	0.816	0.937	0.974
Ours	No	K	-	0.140	1.014	5.473	0.222	0.816	0.937	0.974
Ours	No	K + CS	80	0.139	0.964	5.309	0.215	0.818	0.941	0.977
Garg <i>et al.</i> [14]	Stereo/Pose	K	50	0.169	1.080	5.104	0.273	0.740	0.904	0.962
Zhou <i>et al.</i> [55]	No	K	50	0.201	1.391	5.181	0.264	0.696	0.900	0.966
Yin <i>et al.</i> [52]	No	K + CS	50	0.147	0.936	4.348	0.218	0.810	0.941	0.977
Ours	No	K	50	0.133	0.778	4.069	0.207	0.834	0.947	0.978

563 $0.15, \beta = 0.1, \gamma_1 = \gamma_2 = 0.001, \mu_1 = \mu_2 = 0.1$). Though
 564 we can also learn the optimal weighting using homoscedastic
 565 uncertainty [24], we find that it achieves no better result
 566 than empirically setting the weights. To avoid confusions
 567 for the proposed loss terms, we summarize the training loss
 568 composition in Algorithm 1.

4. Experiments

4.1. Training Dataset

576 **KITTI.** We evaluate our method on the widely-used
 577 KITTI datasets [15, 36]. To conduct fair comparisons with
 578 previous methods, we use the KITTI raw dataset with Eigen
 579 split [9] for evaluating depth estimation, and the KITTI
 580 odometry dataset for pose estimation. Training and testing
 581 images are down-sampled to 128×416 to facilitate the
 582 training and provide a fair evaluation setting. For Eigen
 583 split, we use 20129 images for training and 2214 images
 584 for validation. The 697 testing images are selected by [9]
 585 from 28 scenes. The images in these scenes are excluded
 586 from the training set. The KITTI odometry dataset contains
 587 11 street-view scenes with ground-truth poses. We follow
 588 the previous convention [52, 55] to train the model on se-
 589 quence 00-08 and test on sequence 09-10. We further split
 590 sequence 00-08 to 18361 images for training and 2030 for
 591 validation. Note that the two KITTI datasets may overlap
 592 with each other, the models for evaluating depth and pose
 593 are different even though trained with the same pipeline.

563 **Cityscapes.** We also try pre-training the model on the
 564 Cityscapes [7] dataset since it is shown that starting from
 565 a pre-trained model boosts the performance [55]. The pro-
 566 cess is conducted without adding feature matches for 60k
 567 steps. 88084 images are used for training and 9659 images
 568 for validation.

4.2. Implementation Details

569 **Data preparation.** Since our method utilizes the feature
 570 matches as a weak geometric supervision, we extract SIFT
 571 feature [34] and conduct pairwise matching offline using
 572 SiftGPU [50]. The putative matches are further filtered by
 573 geometric verification using fundamental matrix [20] with
 574 RANSAC [12]. We select 100 pairs of matches for each
 575 image pairs and use them for training. Matches are only
 576 used for training and not necessary for inference.

577 **Learning.** We implement our learning pipeline using Ten-
 578 sorflow [1]. We use training images of size 128×416 , which
 579 is consistent with most of the previous unsupervised learn-
 580 ing approaches [52, 55] to set up a comparable setting. We
 581 use ResNet-50 [21] as the depth encoder as it is proved to
 582 be a more effective model [52]. The depth decoder is a
 583 symmetric architecture that upsamples the encoded depth
 584 feature using transpose convolutions mimicing the inverse
 585 the encoder. The relative pose net is composed of 7 con-
 586 volutional layers (one (7×7) -kernel layer, one (5×5) -
 587 kernel layer and five (3×3) -kernel layers), with the lengths
 588 of feature maps reduced by half and the number of feature
 589 channels multiplied by two from each previous layer. If
 590

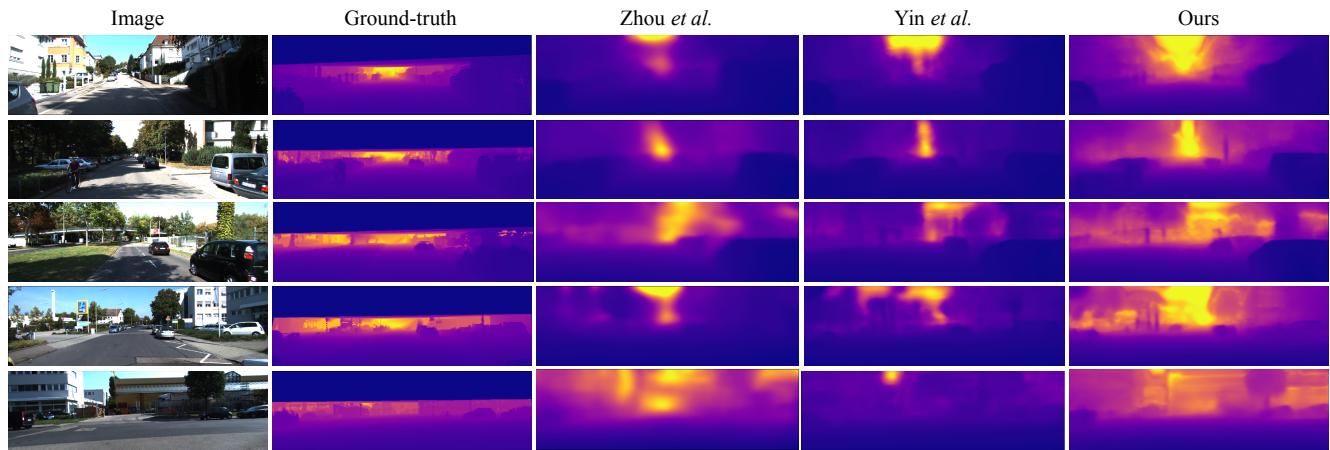


Figure 3. Qualitative comparison for depth estimation on the Eigen split. The predicted depth maps are first aligned with the ground-truth using medians. Then the depth values larger than 80m are set to 80m to ensure a consistent color scale. It shows that our result best reflects the ground-truth depth range and contains richer details (best view in color).

not explicitly specified, we train the neural networks using 3-view image sequences as the photometric error would accumulate for longer input sequences. We use the Adam [27] solver with $\beta_1 = 0.9$, $\beta_2 = 0.999$, a learning rate of 0.0001 and a batch size of 4. We adopt ResNet-50 [21] as the depth encoder and the same architecture for pose network as [55].

4.3. Depth Estimation

We compare our depth estimation performance with state-of-the-art supervised and unsupervised methods. The evaluation metrics protocol follows a series of previous works [35, 52, 55]. As shown in Table 1, our method achieves the best performance among all the methods that jointly learn depth and pose. Previous methods often filter the predicted depth map by setting a maximum depth at 50m or 80m (the ground-truth depth range is within 80m) before computing depth error, since distant pixels may have larger prediction error. We also evaluate the performance without filtering the predicted depth maps. Table 1 shows that without capping the maximum depth [52, 55] become worse while our result seldomly changes, meaning our consistent training renders depth predictions with little noise and this filtering process is unnecessary. Figure 3 provides a qualitative comparison of the predictions. We show the depth value (the nearer the darker) instead of the inverse depth (disparity) parameterization, which makes the distant region stand out.

It is noted that our method achieves worse performance than [16] which is trained and tested on larger images using the rectified stereo images pairs without estimating the relative pose. It is also noteworthy that [28] and ours both use self-supervised weak supervisions. This implies that the raw feature matches are more robust as the supervisory signal, whereas using pose and depth computed from classical

Table 2. Pose estimation evaluation. All the learning-based methods are trained and tested on 128×416 images, while ORB-SLAM2 are tested on full-sized (370×1226) images.

Method	Seq 09	Seq 10
ORB-SLAM2 [37]	0.014 ± 0.008	0.012 ± 0.011
Zhou et al. [55] updated (5-frame)	0.016 ± 0.009	0.013 ± 0.009
Yin et al. [52] (5-frame)	0.012 ± 0.007	0.012 ± 0.009
Mahjourian et al. [35], no ICP (3-frame)	0.014 ± 0.010	0.013 ± 0.011
Mahjourian et al. [35], with ICP (3-frame)	0.013 ± 0.010	0.012 ± 0.011
Klodt et al. [28] (5-frame)	0.014 ± 0.007	0.013 ± 0.009
Ours et al. (3-frame)	0.009 ± 0.005	0.008 ± 0.007

SfM [28] is possible to introduce additional bias inherited from the PnP [31] or triangulation algorithms.

4.4. Pose Estimation

We evaluate the performance of relative pose estimation on the KITTI odometry datasets. We have observed that with the pairwise matching supervision, the result for motion estimation has been extensively improved. We measure the Absolute Trajectory Error (ATE) over N -frame snippets. The mean error and variance is averaged from the full sequence. As shown in Table 2, with the same underlying network structure, the proposed method outperforms state-of-the-art methods by a large margin.

4.5. Ablation Study

Performance with different modules. After horizontally comparing with different architectures, we also evaluate our method vertically to show the effects of different modules. The benchmarks are conducted in terms of both depth and pose trained solely on KITTI raw dataset and KITTI odometry dataset correspondingly. Since there are too many combinations of using different loss terms, we choose an incremental order to add the proposed techniques. When the

756 Table 3. Evaluation of different training loss configurations. All models are either solely trained on KITTI raw dataset (for depth) or KITTI
 757 odometry dataset (for pose) without pre-training on Cityscapes. The depth estimation performance is evaluated with maximum depth
 758 set/capped at 80m. All models except the last two are trained on 3-view image sequences. The best result for each metric is **bolded**.
 759

Baseline	Loss Configuration					Depth (KITTI raw Eigen split)					Pose (KITTI odometry)			
	Epipolar	Re-projection	Forward-backward	Multi-view	Uncertainty	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25^3$	$\delta < 1.25^2$	$\delta < 1.25^3$	Seq 09	Seq 10
✓	-	-	-	-	-	0.163	1.371	6.275	0.249	0.773	0.918	0.966	0.014 ± 0.009	0.012 ± 0.012
✓	✓	-	-	-	-	0.159	1.287	5.725	0.239	0.791	0.927	0.969	0.010 ± 0.005	0.009 ± 0.008
✓	✓	✓	-	-	-	0.152	1.205	5.56	0.227	0.800	0.935	0.973	0.009 ± 0.005	0.009 ± 0.008
✓	✓	✓	✓	-	-	0.146	1.391	5.791	0.229	0.814	0.936	0.972	0.009 ± 0.005	0.008 ± 0.007
✓	✓	✓	✓	✓	-	0.140	1.014	5.473	0.222	0.816	0.937	0.974	0.009 ± 0.005	0.008 ± 0.007
✓	✓	✓	✓	✓	✓	0.143	1.003	5.238	0.216	0.812	0.942	0.978	0.009 ± 0.005	0.008 ± 0.007
✓ (5-view)	-	-	-	-	-	0.169	1.607	6.129	0.255	0.779	0.917	0.963	0.014 ± 0.009	0.013 ± 0.009
✓ (5-view)	✓	✓	-	-	-	0.157	1.449	5.796	0.239	0.803	0.929	0.970	0.012 ± 0.008	0.010 ± 0.007

765
 766 configuration changes, we restart the training from scratch
 767 and the statistics are recorded when the training process is
 768 converged. Based on the results in Table 3, we have the
 769 following observations:
 770

- 771 • The re-implemented baseline model (first line in Table 3),
 772 using the total loss from Equation 5, has already surpassed
 773 several models [28, 35, 55]. The reasons can be
 774 attributed to 1) the strong depth encoder ResNet-50 [21],
 775 the same as in [52]; 2) the SSIM loss which [55] lacks.
 776
- 777 • Combined with the epipolar loss term \mathcal{L}_{epi} , the result
 778 for the relative pose estimation is greatly improved. It
 779 shows the efficacy of using raw feature matches as the
 780 weakly supervised signal. However, the improvement for
 781 depth estimation is not as significant as pose estimation,
 782 because \mathcal{L}_{epi} does not concern the estimated depth.
 783
- 784 • Re-projection loss inferred from sparse feature matches
 785 further improves the monocular depth inference. The im-
 786 provements for pose estimation brought by ingredients
 787 other than the epipolar loss are marginal.
 788
- 789 • The forward-back consistency and multi-view con-
 790 sistency are the essential parts for the improvement in depth
 791 estimation.
 792
- 793 • Adding probabilistic uncertainty mask only improves the
 794 result slightly on some metrics, though it completes the
 795 theoretical foundation by taking data uncertainty into
 796 consideration. This observation is consistent with [55]
 797 that it is possible to be determined by the characteristics
 798 of the training dataset (KITTI).

799 In summary, the proposed weak geometric supervision
 800 helps most for the pose estimation, while the consistency
 801 terms proposed in Section 3.3 essentially improve mono-
 802 cular depth estimation.
 803

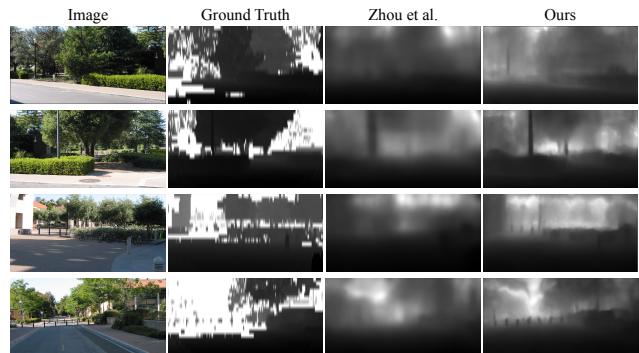
804 **Sequence Length.** The multi-view depth consistency loss
 805 brings boosts for depth estimation. However, the perfor-
 806 mance boost can be also attributed to using longer im-
 807 age snippets, since similar second-order relations can be
 808 exploited by using 5-view image sequences for training.
 809 Therefore, we further evaluate the performance of using

810
 811 Table 4. Generalization experiments on Make3D. The evaluation
 812 metrics are the same as the ones used in Table 1 except the last one
 813 (RMSE \log_{10}) to conform with [23]. The methods marked with †
 814 are supervised by Make3D training depth. The depth estimation
 815 performance is evaluated with maximum depth set/capped at 70m.
 816 We use the center-cropped images as in [16] and resize them to
 817 128 × 416 for inference.
 818

Method	Supervision		Metrics			
	depth	pose	Abs Rel	Sq Rel	RMSE	RMSE \log_{10}
Karsch <i>et al.</i> [23]†	✓	-	0.417	4.894	8.172	0.144
Liu <i>et al.</i> [33]†	✓	-	0.462	6.625	9.972	0.161
Laina <i>et al.</i> [30] †	✓	-	0.198	1.663	5.461	0.082
Godard <i>et al.</i> [16]	-	✓	0.443	7.112	8.860	0.142
Zhou <i>et al.</i> [55]	-	-	0.392	4.473	8.307	0.194
Ours	-	-	0.378	4.348	7.901	0.183

819
 820 5-view training images with different configurations. As
 821 shown in Table 3, training on longer image sequences would
 822 deteriorate the performance, because long sequences also
 823 contain larger photometric noises. We argue that the pro-
 824 posed formulation elevates the results not from more data
 825 terms, but the consistency embedded in geometric relations.
 826

4.6. Generalization on Make3D



827
 828 Figure 4. Sample depth predictions on the Make3D dataset. Both
 829 our method and SfMlearner [55] are trained on Cityscapes+KITTI
 830 datasets.
 831

832 To illustrate that the proposed method is able to general-
 833 ize to other datasets unseen in the training, we compare to
 834 several supervised/self-supervised methods on the Make3D
 835 dataset [43], whose test set contains 134 RGB-D images.
 836 We use the same evaluation protocol in [16] which crops the
 837

864 central region of images for testing. As shown in Table 4, 918
865 our best model, pre-trained on Cityscapes and finetuned on 919
866 KITTI raw dataset, achieves reasonable generalization abil- 920
867 ity and even beats several supervised methods on some met- 921
868 rics. A qualitative comparison is shown in Figure 4. 922
869
870 **5. Conclusion** 923
871
872 We have presented an unsupervised pose and depth esti- 924
873 mation pipeline that absorbs both the geometric principles 925
874 and learning-based metrics. We emphasize on the consis- 926
875 tency issues during the training process and propose novel 927
876 ingredients to make the result more robust and reliable. Our 928
877 method achieves the best performance in terms of depth and 929
878 motion estimation among recent state-of-the-art methods. 930
879 The current methods are still far from solving the SfM prob- 931
880 lem in an end-to-end fashion. Further investigations include 932
881 enforcing consistency across the whole unordered dataset, 933
882 such as loop closure and bundle adjustment techniques, us- 934
883 ing the learning-based methods. 935
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917

918

919

920

921

922

923

924

925

926

927

928

929

930

931

932

933

934

935

936

937

938

939

940

941

942

943

944

945

946

947

948

949

950

951

952

953

954

955

956

957

958

959

960

961

962

963

964

965

966

967

968

969

970

971

972

References

973

974

975

976

977

978

979

980

981

982

983

984

985

986

987

988

989

990

991

992

993

994

995

996

997

998

999

1000

1001

1002

1003

1004

1005

1006

1007

1008

1009

1010

1011

1012

1013

1014

1015

1016

1017

1018

1019

1020

1021

1022

1023

1024

1025

- [1] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al. Tensorflow: a system for large-scale machine learning. In *OSDI*, 2016. 6
- [2] S. Agarwal, N. Snavely, I. Simon, S. M. Seitz, and R. Szeliski. Building rome in a day. In *ICCV*, 2009. 2
- [3] A. Baumberg. Reliable feature matching across widely separated views. In *CVPR*, 2000. 1
- [4] J. R. Bergen, P. Anandan, K. J. Hanna, and R. Hingorani. Hierarchical model-based motion estimation. In *ECCV*, 1992. 3
- [5] M. Bloesch, J. Czarnowski, R. Clark, S. Leutenegger, and A. J. Davison. Codeslam-learning a compact, optimisable representation for dense visual slam. In *CVPR*, 2018. 1
- [6] S. L. Bowman, N. Atanasov, K. Daniilidis, and G. J. Pappas. Probabilistic data association for semantic slam. In *ICRA*, 2017. 1
- [7] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016. 1, 2, 4, 6
- [8] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse. Monoslam: Real-time single camera slam. *PAMI*, 2007. 2
- [9] D. Eigen, C. Puhrsch, and R. Fergus. Depth map prediction from a single image using a multi-scale deep network. In *NIPS*, 2014. 2, 6
- [10] J. Engel, V. Koltun, and D. Cremers. Direct sparse odometry. *PAMI*, 2017. 2, 3
- [11] J. Engel, T. Schöps, and D. Cremers. Lsd-slam: Large-scale direct monocular slam. In *ECCV*, 2014. 2, 3
- [12] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 1981. 6
- [13] Y. Gal. Uncertainty in deep learning. *University of Cambridge*, 2016. 5
- [14] R. Garg, V. K. BG, G. Carneiro, and I. Reid. Unsupervised cnn for single view depth estimation: Geometry to the rescue. In *ECCV*, 2016. 2, 6
- [15] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun. Vision meets robotics: The kitti dataset. *International Journal of Robotics Research (IJRR)*, 2013. 1, 2, 4, 6
- [16] C. Godard, O. Mac Aodha, and G. J. Brostow. Unsupervised monocular depth estimation with left-right consistency. In *CVPR*, 2017. 2, 3, 4, 6, 7, 8
- [17] V. M. Govindu. Robustness in motion averaging. In *ACCV*, 2006. 2
- [18] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003. 1, 4
- [19] R. I. Hartley. In defence of the 8-point algorithm. In *ICCV*, 1995. 4
- [20] R. I. Hartley. In defense of the eight-point algorithm. *PAMI*, 1997. 1, 6
- [21] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1, 6, 7, 8

- [22] M. Jaderberg, K. Simonyan, A. Zisserman, et al. Spatial transformer networks. In *NIPS*, 2015. 1, 3, 4 1026
- [23] K. Karsch, C. Liu, and S. B. Kang. Depth transfer: Depth extraction from video using non-parametric sampling. *PAMI*, 2014. 8 1027
- [24] A. Kendall, R. Cipolla, et al. Geometric loss functions for camera pose regression with deep learning. In *CVPR*, 2017. 6 1028
- [25] A. Kendall and Y. Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *NIPS*, 2017. 5 1029
- [26] S. J. Kim and M. Pollefeys. Robust radiometric calibration and vignetting correction. *PAMI*, 2008. 2 1030
- [27] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 7 1031
- [28] M. Klodt and A. Vedaldi. Supervising the new with the old: learning sfm from sfm. In *ECCV*, 2018. 1, 2, 3, 6, 7, 8 1032
- [29] K. Konolige and M. Agrawal. Frameslam: From bundle adjustment to real-time visual mapping. *IEEE Transactions on Robotics*, 2008. 2 1033
- [30] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab. Deeper depth prediction with fully convolutional residual networks. In *3DV*, 2016. 8 1034
- [31] V. Lepetit, F. Moreno-Noguer, and P. Fua. Epnp: An accurate o (n) solution to the pnp problem. *IJCV*, 2009. 7 1035
- [32] F. Liu, C. Shen, G. Lin, and I. D. Reid. Learning depth from single monocular images using deep convolutional neural fields. *PAMI*, 2016. 2, 6 1036
- [33] M. Liu, M. Salzmann, and X. He. Discrete-continuous depth estimation from a single image. In *CVPR*, 2014. 8 1037
- [34] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004. 6 1038
- [35] R. Mahjourian, M. Wicke, and A. Angelova. Unsupervised learning of depth and ego-motion from monocular video using 3d geometric constraints. In *CVPR*, 2018. 2, 3, 6, 7, 8 1039
- [36] M. Menze and A. Geiger. Object scene flow for autonomous vehicles. In *CVPR*, 2015. 6 1040
- [37] R. Mur-Artal and J. D. Tardós. Orb-slam2: An open-source slam system for monocular, stereo, and rgbd cameras. *IEEE Transactions on Robotics*, 2017. 2, 3, 7 1041
- [38] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison. Dtam: Dense tracking and mapping in real-time. In *ICCV*, 2011. 2 1042
- [39] D. Nistér. An efficient solution to the five-point relative pose problem. *PAMI*, 2004. 1 1043
- [40] M. Poggi, F. Tosi, and S. Mattoccia. Learning monocular depth estimation with unsupervised trinocular assumptions. In *3DV*, 2018. 4 1044
- [41] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. Orb: An efficient alternative to sift or surf. In *ICCV*, 2011. 2 1045
- [42] A. Saxena, S. H. Chung, and A. Y. Ng. Learning depth from single monocular images. In *NIPS*, 2006. 1, 2 1046
- [43] A. Saxena, M. Sun, and A. Y. Ng. Make3d: Learning 3d scene structure from a single still image. *PAMI*, 2009. 8 1047
- [44] T. Shen, S. Zhu, T. Fang, R. Zhang, and L. Quan. Graph-based consistent matching for structure-from-motion. In *ECCV*, 2016. 1 1048

- 1080 [45] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon. Bundle adjustment: a modern synthesis. In *International workshop on vision algorithms*, 1999. 2 1134
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
- [46] B. Ummenhofer, H. Zhou, J. Uhrig, N. Mayer, E. Ilg, A. Dosovitskiy, and T. Brox. Demon: Depth and motion network for learning monocular stereo. In *CVPR*, 2017. 2 1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187