Dense Monocular Depth Estimation in Complex Dynamic Scenes

René Ranftl[1], Vibhav Vineet[1], Qifeng Chen[2], and Vladlen Koltun[1]

[1]Intel Labs
[2]Stanford University

Abstract

We present an approach to dense depth estimation from a single monocular camera that is moving through a dynamic scene. The approach produces a dense depth map from two consecutive frames. Moving objects are reconstructed along with the surrounding environment. We provide a novel motion segmentation algorithm that segments the optical flow field into a set of motion models, each with its own epipolar geometry. We then show that the scene can be reconstructed based on these motion models by optimizing a convex program. The optimization jointly reasons about the scales of different objects and assembles the scene in a common coordinate frame, determined up to a global scale. Experimental results demonstrate that the presented approach outperforms prior methods for monocular depth estimation in dynamic scenes.

Figure 1: Given two frames from a monocular video of a dynamic scene captured by a single moving camera, our approach computes a dense depth map that reproduces the spatial layout of the scene, including the moving objects. Top: input frames. The white vehicle is approaching the camera, while the camera itself undergoes forward translation and in-plane rotation. Bottom: the estimated depth map.

1. Introduction

Can mobile monocular systems densely estimate the spatial layout of complex dynamic scenes? Can a mobile robot, UAV, or wearable device equipped with a single video camera see complex dynamic environments in three dimensions? In static scenes, dense depth can be recovered from a single video captured by a moving camera using the established theory of multiple view geometry [35, 9]. Can this theory be extended to reconstruct complex scenes from monocular video when both the camera and the scene are in motion?

To support challenging field applications, a monocular depth reconstruction approach should have a number of characteristics. It must provide complete dense reconstructions, in order to support dense mapping and detailed geometric reasoning. It must natively handle complex scenes, with dozens of objects moving independently in a complex environment. It must accommodate non-rigid motion, so as to properly perceive people, animals, and articulated structures. And it must accommodate realistic camera models, including perspective projection.

In this paper, we present a monocular depth estimation approach that has all of these characteristics. The approach densely estimates depth throughout the visual field, including both static and dynamic parts of the environment. Multiple moving objects, complex geometry, and non-rigid motion are accommodated. The approach works with perspective cameras and yields metric reconstructions.

Our approach comprises two stages. The first stage performs motion segmentation. This stage segments the dynamic scene into a set of motion models, each described by its own epipolar geometry. This enables reconstruction of each component of the scene up to an unknown scale. We propose a novel motion segmentation algorithm that is based on a convex relaxation of the Potts model [5]. Our algorithm supports dense segmentation of complex dynamic scenes into possibly dozens of independently moving components.

The second stage assembles the scene in a common met-

ric frame by jointly reasoning about the scales of different components and their location relative to the camera. The main insight is that moving objects do not exist in a vacuum, but fulfill intrinsic occluder-occludee relationships with respect to each other and the static environment. This can be used to reason about the placement of different objects in the scene. We formulate this reconstruction problem as continuous optimization over scales and depths and introduce ordering and connectivity constraints to assemble the scene. The result is a reconstruction of the dynamic scene from only two frames, determined up to a single global scale.

We evaluate the presented approach on complex dynamic sequences from the challenging Sintel and KITTI datasets [4, 13]. In all cases, the input is monocular video: we do not use stereo or depth input. Our approach outperforms prior depth estimation techniques by a significant margin. Figure 1 shows a reconstruction produced by the presented approach on a dynamic scene from the KITTI dataset.

2. Prior Work

Three significant families of approaches have been proposed for estimating dynamic scene geometry from monocular video: multibody structure-from-motion, non-rigid structure-from-motion, and non-parametric depth transfer. We briefly review each approach in turn.

Multibody structure-from-motion is the most direct extension of classical multi-view geometry to dynamic environments [10, 24, 21, 38, 26, 29]. This approach is based on the assumption that the environment consists of multiple rigidly moving objects. The basic idea is to cluster feature tracks and fit rigid motion models to each cluster. Since each cluster is assumed to be rigid, traditional multi-view techniques can be applied to estimate its motion, assuming proper segmentation and a sufficient number of tracks. This approach typically assumes a small set of rigid objects in the scene and has not produced detailed reconstructions of complex scenes with non-rigidly moving objects. We contribute new robust formulations that accommodate significantly more general objects and environments.

The second family of approaches for three-dimensional reconstruction of dynamic scenes from monocular video is non-rigid structure-from-motion [33, 1, 28, 31]. The elegant mathematical formulations employed by these approaches hinge on strong assumptions: typically, object shape or motion trajectory matrices are assumed to be low-rank and the camera model is assumed to be orthographic. This severely restricts the applicability of these techniques. While recent work has sought to relax some of the constraints of earlier formulations [12, 27, 11], significant limitations remain. For example, Garg et al. [12] reconstruct a single foreground object that is assumed to be manually presegmented. Russell et al. [27] deal with scenes dominated by a single foreground object and demonstrate reconstruction results qualitatively on three videos. Fragkiadaki et al. [11] produce non-metric reconstructions of track clusters in separate coordinate systems and do not estimate the layout of the scene. In contrast, our approach estimates dense depth for complex dynamic scenes over the entire visual field. All objects are reconstructed jointly, yielding consistent reconstructions of complete scenes.

The third family of approaches for monocular depth estimation in dynamic scenes is non-parametric depth transfer [19, 20]. This approach relies on the availability of a dataset of color-depth image pairs at test time. The dataset is assumed to contain scenes that have a similar geometric layout and similar appearance to the test scene. For a given test video, similar images are retrieved from the dataset for every frame, corresponding depth images are warped to fit the test frames, and the resulting depth estimates are spatio-temporally regularized. This approach requires the availability of an appropriate dataset with ground-truth depth data at test time. It is limited to environments that are compatible with the available training data. In contrast, we present a geometric method that does not require a training dataset and naturally applies to novel environments.

Motion and epipolar models can also be used to improve optical flow estimation. Hornacek et al. [17] used an over-parametrization approach to estimate optical flow, which also explicitly reasons about the depth and rigid body motion at each pixel in the image. They recover depth only up to an unknown scale for each rigid object, since their main goal is to use epipolar models to guide optical flow estimation.

3. Overview and Preliminaries

The proposed pipeline consists of two major stages. First, the scene is segmented into a set of epipolar motion models. The segmentation is performed on optical flow and is formulated as a variational labeling problem. (Note that segmentation of the optical flow field has been explored in the past [36, 30, 37].) The second stage performs triangulation and joint reconstruction of all objects. The key assumption in the second stage is that the scene consists of objects that are connected in space. In particular, we assume that dynamic objects are connected to the surrounding environment. This assumption is true for many scenes likely to be encountered by a mobile vision system, such as a robot or a wearable device. In particular, vehicles and people are generally supported by surrounding structures. Note that we do not make narrow assumptions about the supporting structures, say by estimating the ground plane, but infer the point of attachment flexibly by reasoning about the scene as a whole.

Let M be the number of pixels in the image. We index integer positions on the image grid using the superscript

i: for example, (x^i, y^i) refers to the x and y coordinates of the pixel indexed by i. We use the standard operator $\nabla : \mathbb{R}^M \to \mathbb{R}^{2M}$ to denote the linear operator corresponding to the discrete forward differences in the x and y directions. We denote the standard Euclidean norm by $\|\cdot\|$ and use subscripts whenever a different norm is used. Specifically, we will make use of the following norm:

$$\|p\|_{2,1} = \sum_{i=1}^{M} \sqrt{(p^i)^2 + (p^{i+M})^2}, \quad p \in \mathbb{R}^{2M}. \tag{1}$$

4. Motion Segmentation

The task of the motion segmentation stage is to decompose the dynamic scene into a set of independent rigid motions, each described by a fundamental matrix, together with a per-pixel assignment to these motion models. Note that this approach automatically oversegments non-rigid objects into approximately rigid parts. We estimate the number of independent rigid motions as part of the global optimization to ensure that non-rigid motions are approximated well. To generate metric reconstructions, we assume that the intrinsic camera parameters are known. In this form, the motion segmentation problem is an instance of the more general multiple-model fitting problem. Existing state-of-the-art approaches typically assume sparse correspondences [18, 22, 27] and are thus unable to process dense correspondence fields in reasonable time. We propose a new approach that efficiently handles dense correspondence fields, as produced by dense optical flow estimation. Our approach is most closely related to the discrete energy-based multiple-model fitting approach of Isack and Boykov [18], but operates on soft assignments and models the data association as a continuous convex problem. This allows us to leverage recent advances in convex optimization [6] and enables an efficient GPU-based implementation.

The motion segmentation takes as input a dense optical flow field $f = (f_x, f_y) : f_x, f_y \in \mathbb{R}^M$ between images I_1 and I_2, and produces a soft assignment $u_l \in [0, 1]^M$ of each pixel to either one of L distinct motion models F_l or an additional outlier label $L + 1$. We formulate this as a joint labeling and estimation problem, where we additionally exploit the fact that nearby pixels are likely to belong to the same motion model:

$$(u_l^*, F_l^*) = \arg\min_{u_l, F_l} \sum_{l=1}^{L+1} u_l \cdot g(F_l) + \|W_l \nabla u_l\|_{2,1} \tag{2}$$

$$\text{subject to} \quad \sum_{l=1}^{L+1} u_l^i = 1, \; u_l^i \geq 0 \tag{SPX}$$

$$\forall l.\ \text{rank}(F_l) = 2. \tag{EPI}$$

To measure the fitting error of the motion models with respect to the observed correspondences, we compute the symmetric distance to the epipolar lines [16] for each model $l \in \{1 \dots L\}$:

$$g^i(F_l) = d(x_1^i, F_l\, x_2^i)^2 + d(x_2^i, F_l^\top x_1^i)^2, \tag{3}$$

where $x_1^i = [x^i, y^i, 1]^\top$ are homogeneous coordinates in the first image, $x_2^i = [x^i - f_x^i, y^i - f_y^i, 1]^\top$ denote their corresponding homogeneous coordinates in the second image, and d denotes the Euclidean point-to-line distance. With a slight abuse of notation, we assign a fixed cost $g(F_{L+1}) = \gamma$ to the outlier label. We further estimate occlusions and gross errors in the optical flow using a forward-backward consistency check and fix the assignment of occluded pixels to the outlier label. The smoothness term $\|W_l \nabla u_l\|_{2,1}$ reflects the fact that nearby correspondences are likely to belong to the same motion model. We use a diagonal weighting matrix W_l to enable edge-preserving regularization based on the reference image I_1:

$$W_l = \text{diag}\left(\exp\left(-\beta \|\nabla I_1\|^2\right)\right). \tag{4}$$

The simplex constraint (SPX) ensures that the soft assignments sum to one at each pixel, thus $u_l^i \in [0, 1]$ can be interpreted as the probability that pixel i belongs to the motion model F_l. The matrices $F_l \in \mathbb{R}^{3\times3}$ encapsulate the epipolar geometry of the pixels belonging to segment l. The rank constraints (EPI) ensure that each F_l is a fundamental matrix [16].

Energy (2) is a joint optimization problem in the unknown motion models F_l and the pixel-to-motion-model assignments u_l. The energy is non-convex due to the complex dependence on the fundamental matrices F_l. Even worse, the number of independent motion models, L, is unknown a priori. For a fixed set of motion models, however, energy (2) is convex [5]. We will exploit this property to derive an iterative algorithm to approximately minimize (2).

Let us first consider a simplified example, where the number of independent motions is known a priori. Thus the number of labels $(L+1)$ is fixed. Energy (2) can be approximately optimized using a block coordinate descent strategy that iterates over the following two steps. First, fix the motion models F_l and optimize for the assignment probabilities u. Second, fix the assignment probabilities u and optimize F_l. We use a recently proposed variant of the primal-dual algorithm [6] that employs entropy proximal terms to implicitly represent the simplex constraints (SPX) in order to efficiently solve for the labeling. The re-estimation of the fundamental matrices F_l can be decomposed over the individual models and solved for all L models in parallel. In particular, we exploit the soft assignments u^i to reweigh

(a) (b) (c)

Figure 2: Example result of the motion segmentation stage. (a)-(b) The input image and the input optical flow [25]. (c) Segmentation result. Black pixels are assigned to the outlier label.

individual correspondences:

$$F_l^* = \arg\min_{F_l} \sum_{i=1}^{M} u_l^i \left((x_1^i)^\top F_l (x_2^i)\right)^2 \quad \text{subject to} \quad \operatorname{rank}(F_l) = 2. \tag{5}$$

These subproblems can be approximately solved using a reweighted version of the normalized 8-point algorithm [16, 39]. It is important to note that we do not perform a hard assignment of correspondences to models. Instead, each correspondence (x_1^i, x_2^i) contributes to the estimation of every model F_l according to its inlier probability u_l^i.

In order to discover the number of motion models we opt for a simple greedy bootstrapping strategy, where we make extensive use of the outlier label to mine potential fundamental matrices from the data. We start by mining a small set of candidate motions by iteratively applying the normalized 8-point algorithm in a robust least-median-of-squares (LMedS) framework [32]. Based on this initialization, we solve energy (2) using the previously described alternating minimization approach until no further decrease in energy can be made. We then expand the pool of candidate motions. New models are added by robustly estimating motion models from pixels that have the outlier label as their most probable assignment. Specifically, we robustly fit a motion model to each connected component of the pixels assigned to the outlier label. We further expand the pool by splitting labels with disconnected regions and fitting motion models to these regions if the size of the region is larger than a threshold T. (Note that we do not remove the original models from the set of models.) We again perform alternating minimization based on this new label set and repeat this process until no further decrease in energy can be made. We found that this strategy is generally able to discover the number of models, the models themselves, and their per-pixel assignments within 10 iterations. A summary of the algorithm can be found in Algorithm 1.

Algorithm 1 Motion Segmentation

```
Add all correspondences to set O
Let F = {∅}
repeat                                          ▷ Initialization
    Find F using LMedS on all (x1, x2) ∈ O
    Remove all points which are inliers to F from O
    F ← F ∪ {F}
until |O| ≤ 7

repeat                                          ▷ Motion Segmentation
    Let L = |F|
    repeat                                      ▷ Data association
        Minimize (2) for u_l to get (u_l)^{n+1}
        Update F_l by solving (5)
    until no decrease in energy (2)

    Recover hard assignment û_l using (6)
    for each l = 1 ... L + 1 do                 ▷ Model discovery
        Split û_l into connected components C_j
        for each j = {1 ... J} with |C_j| > T do
            Find F using LMedS on (x1, x2) ∈ C_j
            F ← F ∪ {F}
        end for
    end for
until
```

The result of the motion segmentation stage is a set of epipolar geometries F_l^* as well as membership probabilities u_l^* for each pixel. We obtain the final pixel-to-model associations by extracting the label with maximum probability from u_l^* to get $\hat{u}_l$:

$$\hat{u}_l^i = \begin{cases} 1 & \text{if} \quad l = \max_{l \in \{1,\ldots,L+1\}} (u_l^*)^i \\ 0 & \text{else.} \end{cases} \tag{6}$$

Figure 2 shows an example result of the motion segmentation stage.

5. Reconstruction

While the results of modern optical flow algorithms are reliable for many scenes, they still exhibit artifacts in most cases. The optical flow may be noisy due to properties of the model (e.g., staircasing artifacts in models leveraging first-order smoothness assumptions), motion boundaries are often badly localized (edge bleeding), and some regions might be completely wrong. A reconstruction pipeline that relies

on optical flow needs to be robust to these errors. We use a superpixel-based formulation in order to robustly reconstruct the dynamic scene from optical flow correspondences and the epipolar models estimated in Section 4.

We begin by triangulating each correspondence that was not labeled an outlier by the motion segmentation stage using its associated motion model F_l^*. This yields a set of depth estimates $z_l \in \mathbb{R}^M$. Note that each depth estimate is only valid for pixel i with $\hat{u}_l^i = 1$. We set pixels that belong to the segment with largest support as environment pixels and fix their scales to 1.

We now estimate the relative scales between all segments. This cannot be done without additional prior assumptions as the problem is ill-posed in general. For example, when a plane is seen in the sky, it is generally impossible to tell how large it is or how far it is: it could be a Boeing 737 at a certain distance or a larger 747 that is farther away. To resolve scale ambiguities and assemble the scene, we use a prior assumption that is often appropriate in daily life: objects are supported by their environment. We model this assumption using a combination of two constraints:

1. An ordering constraint, which captures the assumption that dynamic objects occlude the static environment. This can be expressed by requiring the inverse depth of segments belonging to the dynamic objects to be larger or equal to the inverse depth of the environment in their immediate vicinity.

2. A smoothness constraint, which states that jumps in inverse depth between dynamic objects and segments belonging to the environment should be minimized. This constraint connects the dynamic objects with the environment, subject to the ordering constraint.

In order to be robust to outliers in the input data, we formulate these constraints as an energy minimization problem defined on a superpixel graph. Consider a superpixel segmentation of the reference image into K segments. That is, each pixel i is assigned to one of K superpixels. We formally write $i \in P_k$ for the set of pixels belonging to superpixel k and denote the edges in the superpixel graph by $\mathcal{E}$. We use Quickshift [34] to produce a superpixel segmentation and break up superpixels that straddle boundaries in the motion segmentation.

Our goal is to estimate a plane for each superpixel k with parameters $\theta_k = [\theta_k^1, \theta_k^2, \theta_k^3]^\top$ and scales $s = [1, s_2, \ldots, s_L]^\top \in \mathbb{R}_+^L$ for all independently moving objects, subject to the previously described constraints. This is formulated as a convex optimization problem with the following objective:

$$E(s, \theta) = E_{\text{ord}}(\theta) + E_{\text{sm}}(\theta) + E_{\text{fit}}(s, \theta). \tag{7}$$

The following paragraphs define the three terms in this objective.

Ordering constraint. Let $\mathcal{E}_d \subset \mathcal{E}$ denote all pairs of edges in the superpixel graph that connect the static environment to dynamic objects. That is, $(k, h) \in \mathcal{E}_d$ if k is part of the environment and h is part of a dynamic object. Let $A_{P_k} \in \mathbb{R}^{|P_k| \times 3}$ be the matrix that results from vertically stacking all $(x_1^i)^\top$ belonging to segment k. We enforce a hard constraint on the planar reconstructions of these pixels that encapsulates the desired ordering:

$$E_{\text{ord}}(\theta) = \sum_{(k,h) \in \mathcal{E}_d} E_{\text{loc}}(\theta, k, h)$$
$$E_{\text{loc}}(\theta, k, h) = \begin{cases} 0 & \text{if} \quad \max(A_{P_k}\theta_k) \leq A_{P_h}\theta_h \\ \infty & \text{else.} \end{cases} \tag{8}$$

Note that this term is convex as it can be represented as a set of linear inequality constraints of the form

$$\theta_k^\top x_1^i \leq A_{P_h}\theta_h, \quad \forall i \in P_k \tag{9}$$

Smoothness term. We impose smoothness by requiring that planes of neighboring superpixels coincide at their boundary $\mathcal{B}$:

$$E_{\text{sm}}(\theta) = \frac{\lambda}{2} \sum_{(k,h) \in \mathcal{E}} \sum_{(i,j) \in \mathcal{B}_{k,h}} w_{kh} \left(\theta_k^\top x_1^i - \theta_h^\top x_1^j \right)^2. \tag{10}$$

The parameter $\lambda > 0$ controls the overall smoothness of the solution and $w_{k,h}$ steers the smoothness according to superpixel appearance:

$$w_{k,h} = \exp\left(-\kappa \left\| m_k - m_h \right\|^2 \right), \tag{11}$$

where m_k and m_h denote the average color of superpixels k and h, respectively.

Fitting term. The fitting term performs a plane fit to the scaled inverse depth values:

$$E_{\text{fit}}(s, \theta) = \sum_{k=1}^{K} \sum_{i \in P_k} w_i \left| \theta_k^\top x_1^i - \sum_{l=1}^{L} \hat{u}_l^i \frac{s_l}{z_l^i} \right|^2. \tag{12}$$

The interpretation of this term is as follows. For each pixel i, the inverse depth is scaled by the factor s_l. Furthermore, the indicator variable $\hat{u}_l^i$ ensures that only a single reconstruction is active at a given pixel. By summing over all epipolar models we arrive at a reconstruction over the complete image, where individual parts are scaled by s_l. $\theta_k^\top x_1^i$ provides the inverse depth value at pixel i according to the

Figure 3: Results on three frames from the KITTI dataset. For each frame, the figure shows the input color image, the ground-truth depth (GT, inpainted for visualization), results produced by depth transfer (DT) [19], and results produced by our approach.

plane parameters θ_k. Note that the fitting term in isolation is underconstrained, since it does not provide any information on the scales s. Arbitrary scales s can lead to the same minimal fitting energy, as the parameters θ can just be scaled accordingly. The weight w_i reweighs the contribution of each pixel according to its residual error:

$$w_i = \begin{cases} \exp\left(-\gamma \sum_{l=1}^{L} \hat{u}_l^i g_l^i\right) & \text{if} \quad \hat{u}_{L+1}^i = 0 \\ 0 & \text{else.} \end{cases} \tag{13}$$

Energy (7) is convex and poses an optimization problem of moderate size. We use CVX for optimization [15, 14], together with an efficient conic solver [23].

6. Evaluation

We evaluate the presented approach quantitatively and qualitatively on two datasets that depict complex and realistic dynamic scenes: the MPI Sintel dataset [4] and the KITTI dataset [13]. To the best of our knowledge, this is the first quantitative evaluation of monocular reconstruction on dynamic scenes of this complexity.

The accuracy of the presented approach is compared to two state-of-the-art techniques for monocular depth estimation from video. The first is the depth transfer approach of Karsch et al. [19], a nonparametric method that relies on training data with ground-truth depth. Due to its nonparametric nature, this approach can be expected to perform well only if the test images are sufficiently similar to images in the training database.

The second approach we compare to is the non-rigid structure-from-motion formulation of Fragkiadaki et al. [11]. This formulation was shown to outperform prior non-rigid structure-from-motion techniques. Unlike our approach, this method produces depth estimates for disconnected tracks, rather than for all pixels. The generated point clouds also lack absolute depth: all depth estimates lie in $[-1, 1]$. To maximize the accuracy reported for this approach, we only measure error along the tracks, rather than over all pixels in the input images. The generated point cloud is scaled to the range of the ground-truth depth map in each frame.

Accuracy is reported using three standard measures. The first is mean relative error (MRE). Let z be the estimated depth and let z_{gt} be the ground-truth depth. MRE is defined as

$$\text{MRE}(z) = \frac{1}{M} \sum_{i=1}^{M} \frac{|z^i - z_{gt}^i|}{z_{gt}^i}. \tag{14}$$

This measures the relative per-pixel error: an error of 0.1m at a depth of 1m is penalized equally to an error of 1m at a depth of 10m. MRE is closely related to the depth contrast measure used to evaluate the effectiveness of different depth cues in human vision [8]. For completeness, we also report the root mean square error (RMSE), defined as $\sqrt{\sum_i (z^i - z_{gt}^i)^2 / M}$, and the $\log_{10}$ error, defined as $\sum_i |\log_{10}(z^i) - \log_{10}(z_{gt}^i)|$. Only pixels with ground-truth depth within 20 meters are used for evaluation. In order to allow for a comparison in terms of absolute depth, we fit a global scale for each method and each frame such that the MRE is minimized.

The runtime of our approach is dominated by the motion segmentation stage, which is in turn dependent on the

complexity of the motion in the scene. We implemented the labeling step on the GPU. All other parts are implemented in Matlab, which results in an execution time that is on the order of 1 minute per frame.

KITTI. We first evaluate the presented approach on the KITTI dataset. Specifically, we use the KITTI odometry set [13]. The dataset provides sparse ground-truth depth measurements acquired by a LiDAR scanner. These are used for quantitative evaluation. We compared the presented approach to the prior approaches introduced above. Unfortunately, the implementation of Fragkiadaki et al. [11] crashes on all sequences in this dataset. We thus report results for depth transfer (DT) [19]. We make the 11 training sequences available to DT at test time. Figure 3 shows qualitative results. Quantitative results are provided in Table 1. DT performs well on this dataset. This can be attributed to the significant resemblance of scenes in the KITTI test set to images in the training data. In particular, the geometric layout of many frames in the KITTI dataset is almost identical. Nevertheless, our approach achieves a higher accuracy than DT on all reported metrics, without relying on any training data.

	MRE	$\log_{10}$	RMSE
Depth Transfer [19]	0.171	0.076	2.830
Ours	**0.148**	**0.065**	**2.408**

Table 1: Quantitative evaluation on the KITTI dataset.

MPI Sintel. The MPI Sintel dataset consists of complex computer graphics sequences. It was constructed for thorough evaluation of optical flow techniques, but has also been used for evaluating other low-level vision algorithms. The advantage of using computer graphics is the availability of precise ground truth. We use monocular sequences of color images as input and report results on the challenging 'final' rendering pass of this benchmark. To maximize the accuracy reported for depth transfer [19], we performed cross-validation such that for each test sequence all other sequences are made available as the training database. We exclude sequences from the evaluation that show no or only insignificant camera motion (*alley_1*, *bandage_1*, *bandage_2*, *shaman_2*).

Table 2 provides the results of a quantitative evaluation on this dataset. To assess the sensitivity of our approach to the input optical flow, we have evaluated the approach when the input flow fields are computed by LDOF [3], EpicFlow [25], and FlowFields [2], respectively. The results demonstrate that the presented approach substantially outperforms the prior work with any of these input flows. With input flows provided by FlowFields [2], the presented approach reduces the MRE by 40% relative to DT [19] and by 30% relative to NR-SfM [11]. The poor performance of DT on this dataset can be explained by the diversity of the sequences. A qualitative comparison is shown in Figure 5.

	MRE	$\log_{10}$	RMSE
Depth Transfer [19]	0.491	0.227	3.334
NR-SfM [11]	0.422	0.231	3.206
Ours – LDOF	0.341	0.154	2.576
Ours – EpicFlow	0.300	0.148	2.575
Ours – FlowFields	**0.297**	**0.146**	**2.458**

Table 2: Quantitative evaluation on the MPI Sintel dataset. We evaluate the accuracy of our approach when different optical flow estimation algorithms are used to produce the input flow. Results are reported on the challenging 'final' rendering pass.

Limitations. The presented formulation is motivated in part by impressive recent advances in optical flow estimation. These advances are ongoing and are expected to continue [7]. Our method can directly benefit from novel optical flow algorithms. On the other hand, if optical flow estimation fails, the presented approach will fail. A number of other limitations are inherent in the purely geometric nature of our approach, which does not use prior information about object shapes and sizes. In particular, the presented formulation will not yield accurate results for objects that are disconnected from their environment, such as birds in flight. Figure 4 shows two failure cases.

(a) (b)

Figure 4: Failure cases. (a) The flying dragon is pushed to the background, which overestimates its depth. (b) Failure due to erroneous input flow.

7. Conclusion

We presented an approach to dense depth estimation from monocular video. Our approach leverages optical flow to segment a dynamic scene into a set of independently moving objects. We reason about the layout of the environment and the placement of the moving objects in it. The

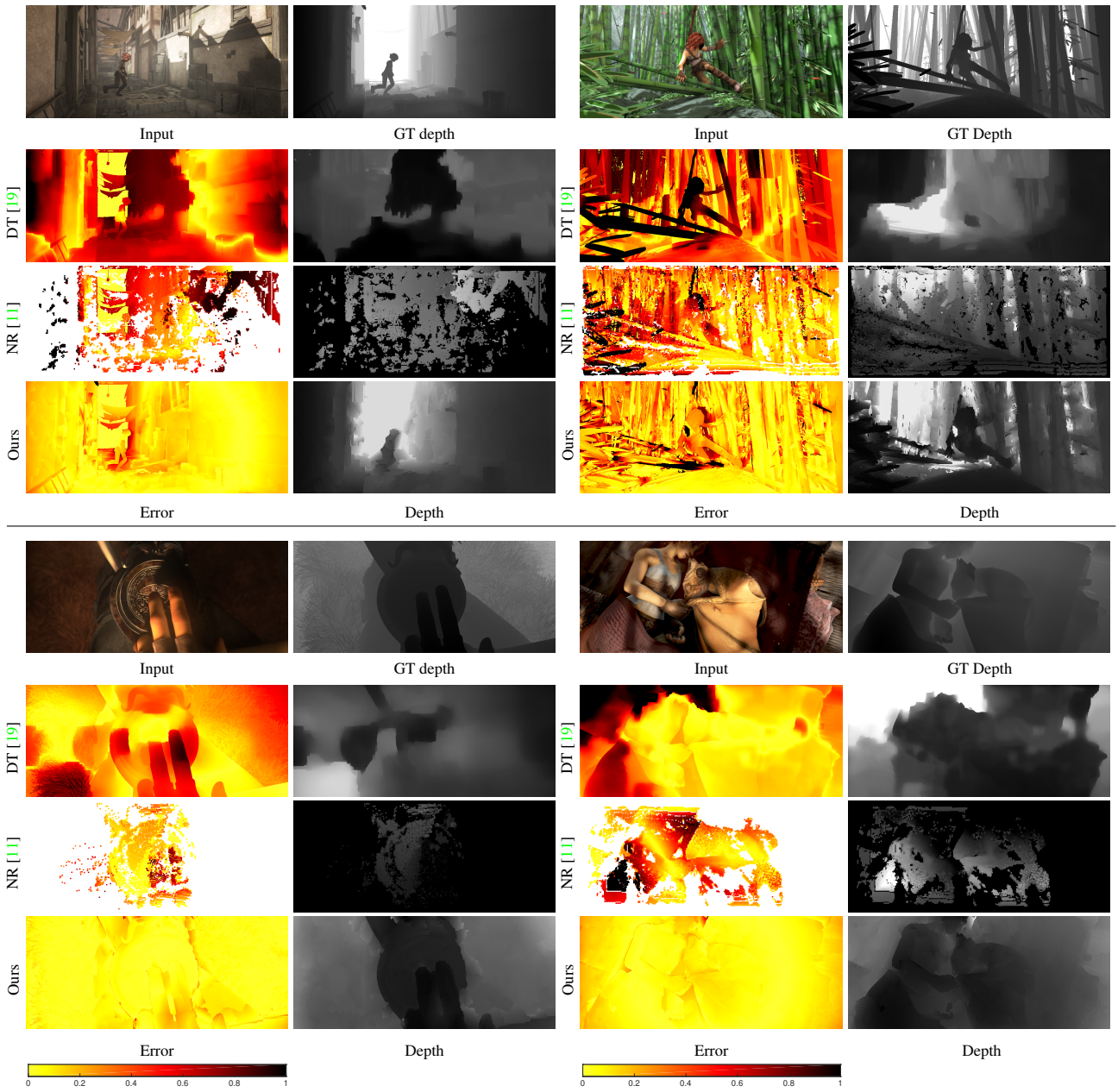

Figure 5: Results on four frames from the MPI Sintel dataset. For each frame, the figure shows the input color image, the ground-truth depth, and results produced by three techniques: depth transfer (DT) [19], non-rigid SfM (NR) [11], and our approach. We dilate the results of NR for visualization. For each technique, the estimated depth map is visualized on the right and per-pixel relative error is visualized on the left.

approach produces dense depth maps of complex dynamic scenes purely from geometric principles.

An important direction for future work is the incorporation of additional prior knowledge into our geometric framework. Nonparametric or learning-based approaches can be leveraged to improve the reconstruction and can also be used to estimate the absolute scale of the scene. We believe that combining these complementary techniques with our geometric approach can lead to a powerful general framework for monocular depth estimation from video. Other opportunities for future work are to couple the optical flow estimation and multiple model fitting and to enforce temporal consistency.

References

[1] I. Akhter, Y. Sheikh, S. Khan, and T. Kanade. Trajectory space: A dual representation for nonrigid structure from motion. *PAMI*, 33(7), 2011. 2

[2] C. Bailer, B. Taetz, and D. Stricker. Flow fields: Dense correspondence fields for highly accurate large displacement optical flow estimation. In *ICCV*, 2015. 7

[3] T. Brox and J. Malik. Large displacement optical flow: descriptor matching in variational motion estimation. *PAMI*, 33(3), 2011. 7

[4] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In *ECCV*, 2012. 2, 6

[5] A. Chambolle, D. Cremers, and T. Pock. A convex approach to minimal partitions. *SIAM Journal on Imaging Sciences*, 5(4), 2012. 1, 3

[6] A. Chambolle and T. Pock. On the ergodic convergence rates of a first-order primal-dual algorithm. *Mathematical Programming*, 2015. 3

[7] Q. Chen and V. Koltun. Full flow: Optical flow estimation by global optimization over regular grids. In *CVPR*, 2016. 7

[8] J. E. Cutting and P. M. Vishton. Perceiving layout and knowing distances: The integration, relative potency, and contextual use of different information about depth. In *Perception of Space and Motion*. Academic Press, 1995. 6

[9] M. Faessler, F. Fontana, C. Forster, E. Mueggler, M. Pizzoli, and D. Scaramuzza. Autonomous, vision-based flight and live dense 3D mapping with a quadrotor micro aerial vehicle. *Journal of Field Robotics*, 2015. 1

[10] A. W. Fitzgibbon and A. Zisserman. Multibody structure and motion: 3-D reconstruction of independently moving objects. In *ECCV*, 2000. 2

[11] K. Fragkiadaki, M. Salas, P. A. Arbeláez, and J. Malik. Grouping-based low-rank trajectory completion and 3D reconstruction. In *NIPS*, 2014. 2, 6, 7, 8

[12] R. Garg, A. Roussos, and L. de Agapito. Dense variational reconstruction of non-rigid surfaces from monocular video. In *CVPR*, 2013. 2

[13] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun. Vision meets robotics: The KITTI dataset. *IJRR*, 32(11), 2013. 2, 6, 7

[14] M. Grant and S. Boyd. Graph implementations for nonsmooth convex programs. In *Recent Advances in Learning and Control*. 2008. 6

[15] M. Grant and S. Boyd. CVX: Matlab software for disciplined convex programming, version 2.1. http://cvxr.com/cvx, 2014. 6

[16] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000. 3, 4

[17] M. Hornacek, F. Besse, J. Kautz, A. Fitzgibbon, and C. Rother. Highly overparameterized optical flow using PatchMatch belief propagation. In *ECCV*, 2014. 2

[18] H. Isack and Y. Boykov. Energy-based geometric multi-model fitting. *IJCV*, 97(2), 2012. 3

[19] K. Karsch, C. Liu, and S. B. Kang. Depth transfer: Depth extraction from video using non-parametric sampling. *PAMI*, 36(11), 2014. 2, 6, 7, 8

[20] N. Kong and M. J. Black. Intrinsic depth: Improving depth transfer with intrinsic images. In *ICCV*, 2015. 2

[21] A. Kundu, K. M. Krishna, and C. V. Jawahar. Realtime multibody visual SLAM with a smoothly moving monocular camera. In *ICCV*, 2011. 2

[22] L. Magri and A. Fusiello. T-linkage: A continuous relaxation of j-linkage for multi-model fitting. In *CVPR*, 2014. 3

[23] B. O'Donoghue, E. Chu, N. Parikh, and S. Boyd. Conic optimization via operator splitting and homogeneous self-dual embedding. *Journal of Optimization Theory and Applications*, 2016. 6

[24] K. E. Ozden, K. Schindler, and L. J. V. Gool. Multibody structure-from-motion in practice. *PAMI*, 32(6), 2010. 2

[25] J. Revaud, P. Weinzaepfel, Z. Harchaoui, and C. Schmid. EpicFlow: Edge-preserving interpolation of correspondences for optical flow. In *CVPR*, 2015. 4, 7

[26] A. Roussos, C. Russell, R. Garg, and L. de Agapito. Dense multibody motion estimation and reconstruction from a handheld camera. In *ISMAR*, 2012. 2

[27] C. Russell, R. Yu, and L. de Agapito. Video pop-up: Monocular 3D reconstruction of dynamic scenes. In *ECCV*, 2014. 2, 3

[28] M. Salzmann and P. Fua. *Deformable Surface 3D Reconstruction from Monocular Images*. Synthesis Lectures on Computer Vision. Morgan & Claypool Publishers, 2010. 2

[29] S. Song and M. Chandraker. Joint SFM and detection cues for monocular 3D localization in road scenes. In *CVPR*, 2015. 2

[30] D. Sun, J. Wulff, E. B. Sudderth, H. Pfister, and M. J. Black. A fully-connected layered model of foreground and background flow. In *CVPR*, 2013. 2

[31] J. Taylor, A. D. Jepson, and K. N. Kutulakos. Non-rigid structure from locally-rigid motion. In *CVPR*, 2010. 2

[32] P. H. S. Torr. Geometric motion segmentation and model selection. *Philosophical Transactions of the Royal Society A*, 356(1740), 1998. 4

[33] L. Torresani, A. Hertzmann, and C. Bregler. Nonrigid structure-from-motion: Estimating shape and motion with hierarchical priors. *PAMI*, 30(5), 2008. 2

[34] A. Vedaldi and S. Soatto. Quick shift and kernel methods for mode seeking. In *ECCV*, 2008. 5

[35] G. Vogiatzis and C. Hernández. Video-based, real-time multi-view stereo. *Image and Vision Computing*, 29(7), 2011. 1

[36] J. Y. A. Wang and E. H. Adelson. Representing moving images with layers. *IEEE Transactions on Image Processing*, 3(5), 1994. 2

[37] J. Wulff and M. J. Black. Efficient sparse-to-dense optical flow estimation using a learned basis and layers. In *CVPR*, 2015. 2

[38] G. Zhang, J. Jia, and H. Bao. Simultaneous multi-body stereo and segmentation. In *ICCV*, 2011. 2

[39] Z. Zhang. Determining the epipolar geometry and its uncertainty: A review. *IJCV*, 27(2), 1998. 4