Robust Consistent Video Based on Monodepth Estimation From Single Image

Group 15

R11921074 劉陶銘、R11921075 江承祐、R11921072 謝子涔

1. Introduction

Dense per-frame depth is an important intermediate profile for many video-based applications, such as self-driving cars utilizing the depth profile to estimate the distance of obstacles, or filtering video with different special effects according to their depths. However, estimating depth in a casually captured video is challenging. Cell phones contain small image sensors that may result in noisy images. The traditional Structure from motion method (SfM) may suffer from these challenges when estimating those videos.

The main spirit of the architecture is utilizing a monocular depth estimation model to obtain an initial depth profile. Then, a series of fine-tuning methods are implemented in the test time stage. Unlike the method CVD which was proposed in 2020, this method does not require COLMAP for camera poses. This improves the robustness of the system, since COLMAP usually fails when motion blur, noise, shake, and rolling shutter deformations take place in the input video. Therefore, Robust CVD can prevent failure that could occur in traditional Structure from Motion methods with the monocular estimation model, MiDas. Furthermore, we further improve the performance of the architecture by replacing the original MiDaS model to the state-of-the-art monocular estimation model, Adelai Depth.

2. Methodology

2.1 Overview

Robust Consistent Video Depth Estimation algorithm only takes a monocular colored video as input. We estimate the initial depth using two distinctive depth estimation models, MiDaS and Adelai Depth. Then, we jointly optimize camera poses as well as the grid wise scale factor (i.e. deformation field). These operations enable us to align the depth maps in 3D and resolve coarse misalignments. With the information provided by the mentioned optimizations, we can generate a geometrically aware depth filter to obtain geometric consistency of the depth profiles. The data processing procedure is depicted as follows.

2.2 Preprocessing

Instead of sampling every frame pairs in the video, we only sample frame pairs with a specific interval.

$$P = \left\{ (i,\ j) \,\middle|\, |i-j| = k,\ i \bmod k = 0,\ k = 1,2,4,\ldots \right\}.$$

We use RAFT[11] to compute a dense optical flow fieldfij for mapping a pixel in frame i to its corresponding location in frame j. We also use mask [9] to exclude potentially dynamic objects (such as pixels in the category “person”,”animal”, or “vehicle”). For flow computation, we downsize the image to make the long side of the image 384 pixels.

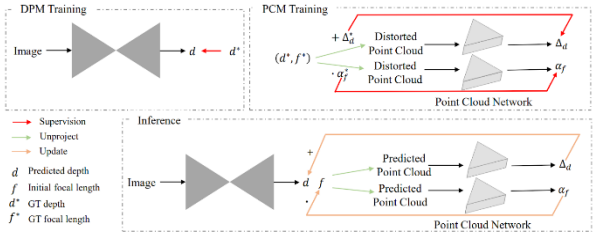
2.3 Estimating Initial Depth

2.3.1 MiDas

In Robust CVD [2], choose MiDas as the top depth estimation model. MiDas is a state-of-art monocular depth estimation model using ResNeXt-101 pretrained with a massive corpus of weakly-supervised data (WSL) as its backbone. It aims at performing across diverse environments, covering from DIML Indoor, MegaDepth, 3D Movies dataset, and so on. To satisfy the diverse genres of datasets, they combine multiple losses to make the model flexible enough to deal with diverse datasets, including Scale- invariant loss, shift-invariant loss and related loss functions. Utilizing the experimental protocol of zero-shot cross-dataset transfer, MiDas has a high reliability of estimating depth from different types of datasets. Here, we would compare the original Robust CVD method with MiDas as the initial depth prediction model.

2.3.2 Adelai

In order to obtain a more precise initial depth estimation, we use Adelai Depth model to generate initial depth. Adelai depth is a monocular depth estimation model which aims to reconstruct 3D point clouds by a monocular image. Adelai Depth also uses ResNeXt-101 as their backbone. During training, the depth prediction model and point cloud module are trained separately on different sources of data. Then, two networks are combined with each other during the inference stage. The depth model estimates depth for the point cloud module, the point cloud module reconstructs the distorted point clouds and compares with the ground truth to obtain the shift and focal length. The information is subsequently fed back to the depth estimation model. This method enables the monocular depth estimation model to be aware of the 3D structure of such input images, which is beneficial for our architecture. This feature is the main reason that we replace MiDaS with Adelai Depth.



2.4 Pose optimization

One of the traditional methods to estimate poses is Structure from Motion. However, structure from motion requires camera intrinsic parameters. Moreover, the depth estimation method that relies on such method has an undesired degradation when pose estimation is not precise. It would be desirable to have a more reliable way to obtain poses for our application than with SfM. In our project, we reverse the role of depth and camera parameters. We assume that we know the depth profile (from the output of the initial depth model), and we optimize camera parameters (i.e R,t,K,s). This modified equation resembles the triangulation in bundle adjustment, but it is more robust since the match image points do not need to be estimated since the depth profile is already known. However, this time it is depth that is required to be precise.

2.5 Deformation Field

The Robust Consistent Video Depth Estimation abandoned the idea presented in their previous work CVD that calculates a FIXED scale factor si over an entire frame, then take the mean of scale factors of all frames to get a video wise scale factor for all depth maps. Such an approach to 3D alignment in CVD is fine since the fine-tuning information relies strongly on the result of COLMAP. In Robust Consistent Video Depth Estimation, however, as mentioned before, relies on the accuracy of the initial depth estimation results. Such an approach to an approximate scale factor which causes the misalignment error will further degrade the estimated poses, which in turn transforms into erroneous depth estimation.

To break through this dilemma is to improve the depth alignment in pose estimation. Robust CVD introduced a spatial varying scale factor, namely deformation field. They replaced the depth scale coefficients with a spatially varying bilinear spline.

Each test time training stage, they estimate the scale factor of the depth map in different sizes of grids. (i.e from 1*1 to 17*10). Moreover, to encourage smooth deformation fields, they add a loss that penalizes large difference scale factors in neighboring grid values.

$$\mathcal{L}^{deform} = \sum_i \sum_{(k,r)\in N} \left\| s_i^k - s_i^r \right\|_2^2 \max(w_i^k, w_i^r)$$

N means the set of all vertically and horizontally neighboring grids, the weights encourage more smoothness in dynamically masked regions.

$$w_i^k = \lambda_1 + \lambda_2 \sum_p m_i^{dyn}(p)\, b_k(p)$$

,where $\lambda_2 >> \lambda_1$

2.6 Depth Filtering

The pose optimization and flexible deformation field we constructed beforehand removes any large-scale misalignments. With the pose of the frames known, the geometry-aware filter can transform the depths from other frames using the reprojection techniques, shown below.

Let p be a 2D pixel coordinate. We can lift it to 3D coordinate by providing the depth profile. p denotes the homogeneous pixel coordinate

$$c_i(p) = s_i\, d_i(p)\, \tilde{p}.$$

If we want to project the 3D point obtained above, we could use the equation below into another frame j

$$c_{i\to j}(p) = K_j R_j^{\top} \left(R_i K_i^{-1} c_i(p) + t_i - t_j \right)$$

The depth filter utilizes the technique and generates the final depth profile:

$$d_i^{final}(p) = \sum_{q\in N(p)} \sum_{j=i-\tau}^{i+\tau} z_{j\to i}\left(\tilde{f}_{i\to j}(q)\right) w_{i\to j}(q).$$

z refers to the scalar z-component of $c_{j\to i}$ on the equation above (i.e. the reprojected depth), f is the flow between frames obtained by chaining flow maps between consecutive frames, w is a data driven weights trained by penalizing low similarity of frame pairs, the implementation details can be discovered in the citation[2].

3. Preprocessing

3.1 Test Data

We choose MPI Sintel[10] as our testing dataset, which consists of 23 synthetic sequences of highly dynamic scenes. The high-level dynamic dataset satisfies our purpose of dealing with dynamic objects without failure. Each sequence contains ground truth depth in meters. With the accurate ground truth annotations, we can demonstrate a quantitative evaluation of estimated depth within both MiDas method and our method.

The title (Times New Roman14-point bold), authors' names (Times New Roman 12-point) and affiliations (Times New Roman 12-point italics) run across the full width of the page. We also recommend e-mail addresses for all authors. See the top of this page for three addresses. If only one address is needed, center all address text. For two addresses, use two centered columns, and so on. If more than three authors, you may have to improvise.

3.2 Metrics

We present per-frame evaluations on standard error and accuracy metrics, including Absolute and Relative Error

(Abs-Rel), Squared Relative Error (Sq-Rel), Root Mean Squared Error (RMSE), and Root Mean Squared Log Error (RMSE-Log). About accuracy, we adopted the delta t to the accuracy of the estimated depth with different thresholds.

TABLE IV:
Standard depth evaluation metrics. The **pred** and **gt** denotes predicted depth and ground truth, respectively. **D** represents the set of all predicted depths value for a single image and $|\,.\,|$ returns the number of the elements in each input set

Abs-Rel	$\frac{1}{\lvert D\rvert}\sum_{pred\in D}\lvert gt-pred\rvert/gt$
Sq-Rel	$\frac{1}{\lvert D\rvert}\sum_{pred\in D}\lVert gt-pred\rVert^2/gt$
RMSE	$\sqrt{\frac{1}{\lvert D\rvert}\sum_{pred\in D}\lVert gt-pred\rVert^2}$
RMSE-Log	$\sqrt{\frac{1}{\lvert D\rvert}\sum_{pred\in D}\lVert \log(gt)-\log(pred)\rVert^2}$
δt	$\frac{1}{\lvert D\rvert}\lvert\{pred\in D\,\lvert max(\frac{gt}{pred},\frac{pred}{gt})<1.25^t\}\rvert\times 100\%$

3.3 Compared Methods

We conduct a comparison between Robust CVD method and our method by testing Sintel dataset. Since the scale of depth estimation is different from the ground truth of Sintel dataset, which are all in meters, we do normalization of min-max feature scaling to estimate and ground truth.

$$X' = \frac{X - X_{\min}}{X_{\max} - X_{\min}}$$

Therefore, we can ensure the error and accuracy calculation won't be influenced by the different scale.

3.3.1 Comparison of initial depth *estimation*

We compared the estimation result with both Robust CVD and our method.

	Depth - Error metric ↓				Depth - Accuracy metric ↑		
	Abs Rel↓	Sq Rel↓	RMSE↓	log RMSE↓	δ < 1.25↑	δ < 1.25^2↑	δ < 1.25^3
MiDas ini	**0.6912**	**0.0855**	**0.1631**	0.7900	0.2574	0.5077	0.6496
Adelai ini	1.87919	0.36391	0.19917	0.59694	0.31294	0.43203	0.77229

For depth estimation images, we can see that the contrast in MiDas is higher than in Adelai. However, the background in MiDas is blurrier than in Adelai.

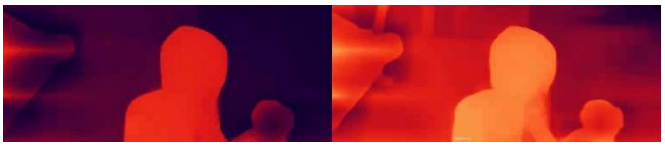
Comparing the depth estimation metrics in MiDas and in Adelai, we can find that the error metric in MiDas is lower than in Adelai. However, the accuracy metric in Adelai is higher than in MiDas.

We can see that the visualization of initial depth estimation in Adelai is better than in MiDas.

3.3.2 Comparison of Final Depth Estimation

After fine-tuning, we suppose that the algorithm can maintain a geometric consistency over frames. However, we can see that the background estimation of MiDas is totally unrecognized. We assume that MiDas can strengthen the contrast of object and background but it also sacrifices the resolution of the overall image. On the contrary, the background is still clear and we can recognize the rough contour in Adelai result. It is the advantage that Adelai Depth model still retains the integrity from the original image. Therefore, in the comparison table, we can see that Adelai Depth performs much better than MiDas in terms of accuracy metric.

	Depth - Error metric ↓				Depth - Accuracy metric ↑		
	Abs Rel↓	Sq Rel↓	RMSE↓	log RMSE↓	δ < 1.25↑	δ < 1.25^2↑	δ < 1.25^3↑
MiDas	0.9402	0.1328	0.2257	0.8891	0.1407	0.3796	0.5138
Ours	1.5344	0.2589	0.1907	**0.5396**	0.3422	0.6454	0.9121



However, the error metric does not outperform the state-of-the-art depth estimation algorithms, we assume that our algorithm pursues the consistency of the entire video, sacrificing the pixel wise accuracy of the depth profile. This in turn makes every pixel contribute a certain amount of error compared to the ground truth, but not big enough to exceed the threshold of the accuracy metric.

We may conclude that the estimated depth map does not guarantee to provide the most accurate depth pixel wise but provide consistency framewise.

4. Preprocessing

4.1 Downsampling

When computing the flow profile of frame pairs, we downsized the frames to reduce the runtime. The drawback is that we sacrifice the resolution to save time. Therefore, we assume that optimizing the algorithm of optical flow could be a method to compute the original frames' optical flow without downsizing it. This might make a noticeable improvement in the accuracy of the estimated depth profile.

4.2 Error and Accuracy

From the metrics shown above, we can see that our algorithm does not outperform the existing methods at all metrics. Further improving the accuracy of the algorithm is a crucial improvement in the future.

4.3 Execution Time

We found that most of the computational time is dedicated to flow reconstruction. Hence, improving the algorithm in computing flow is a good direction. For example, we can sample frame pairs and compute flow concurrently when estimating initial depth profile.

References

[1] Luo, et al. Consistent video depth estimation. SIGGRAPH, 2020

[2] Kpof, et al. Robust Consistent Video Depth Estimation. CVPR, 2021

[3] Yin, et al. Learning to Recover 3D Scene Shape from a Single Image. CVPR, 2020

[4] Ranftl, et al. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. TPAMI, 2020.

[5] Masoumian,et al. GCNDepth: Self-supervised monocular depth estimation based on graph convolutional network.T-ITS,2021

[6] Xu, et al. Towards 3D Scene. CVPR, 2021

[7] Liu, et al. Pointvoxel cnn for efficient 3d deep learning. In Proc. Advances in Neural Inf. Process. Syst., 2019

[8] Sameer Agarwal, Keir Mierle, and Others. Ceres solver 2.1, 2022 http://ceres-solver.org/

[9] He, Kaiming, et al. "Mask r-cnn." Proceedings of the IEEE international conference on computer vision. 2017.

[10] Butler, Daniel J., et al. "A naturalistic open source movie for optical flow evaluation." European conference on computer vision. Springer, Berlin, Heidelberg, 2012.

[11] Teed, Zachary, and Jia Deng. "Raft: Recurrent all-pairs field transforms for optical flow." European conference on computer vision. Springer, Cham, 2020.