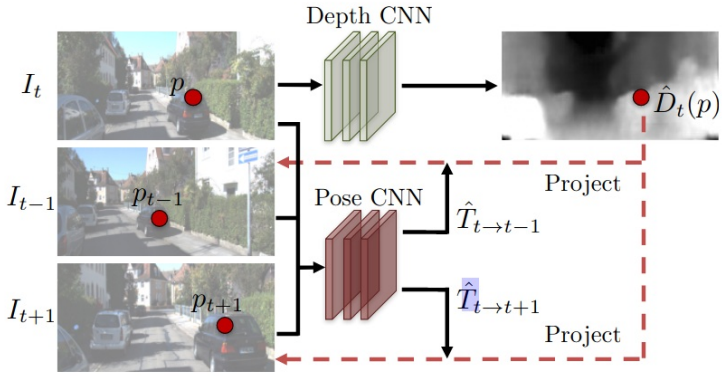
The Temporal Opportunist: Self-Supervised Multi-Frame Monocular Depth [Manydepth (CVPR2021)]

Keyword: monocular depth estimation, self-supervised training Github Link: Manydepth

Intro to the development of self-supervised monocular depth estimation (UCL Gabriel Brostow)

1. View synthesis supervision [SfMLearner (CVPR2017)]

- Combine depth estimation with pose estimation using view synthesis as supervision
- A depth estimation module (encoder, decoder) and a pose estimation module (encoder, decoder)



Overview of the supervision pipeline based on view synthesis

$$
\begin{aligned}
&D_t \cdot p_t = K \cdot P_t \\
&P_t = K^{-1} \cdot D_t \cdot p_t \\
&P_{t+n} = T_{t\to t+n} \cdot P_t \\
&\tilde{p_{t+n}} = K \cdot P_{t+n} = K \cdot T_{t\to t+n} \cdot K^{-1} \cdot D_t \cdot p_t \\
&Loss = L1(\tilde{p_{t+n}}, p_{t+n})
\end{aligned}
$$

Reprojection formula

- Drawbacks: loss is naive (L1 loss between source pixel and the warpped/reprojected pixel)

2. Appearance matching loss [Monodepth1 (CVPR2017)]

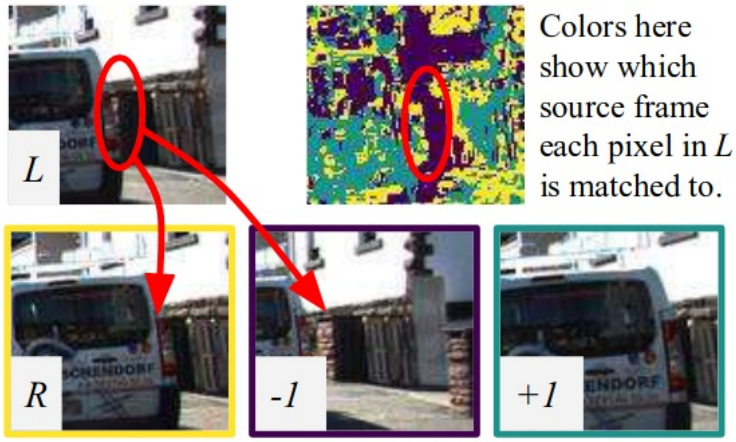
- Key idea: combine L1 loss with structural similarity loss (SSIM)

$$C_{ap}^{l} = \frac{1}{N}\sum_{i,j} \alpha \frac{1-\mathrm{SSIM}(I_{ij}^{l}, \tilde{I}_{ij}^{l})}{2} + (1-\alpha)\left\| I_{ij}^{l} - \tilde{I}_{ij}^{l} \right\|.$$

- Drawbacks: stereo data in training process are required

3. Per-pixel minimum reprojection loss [(Monodepth2 (ICCV2019))]

- Key idea: select the minimum reprojection loss to avoid the influence by occlusion



$$L_p = \min_{t'} pe(I_t, I_{t' \to t}).$$

Overview of the minimum reprojection loss

4. Auto-masking stationary pixels [([Monodepth2 (CVPR2019)])]

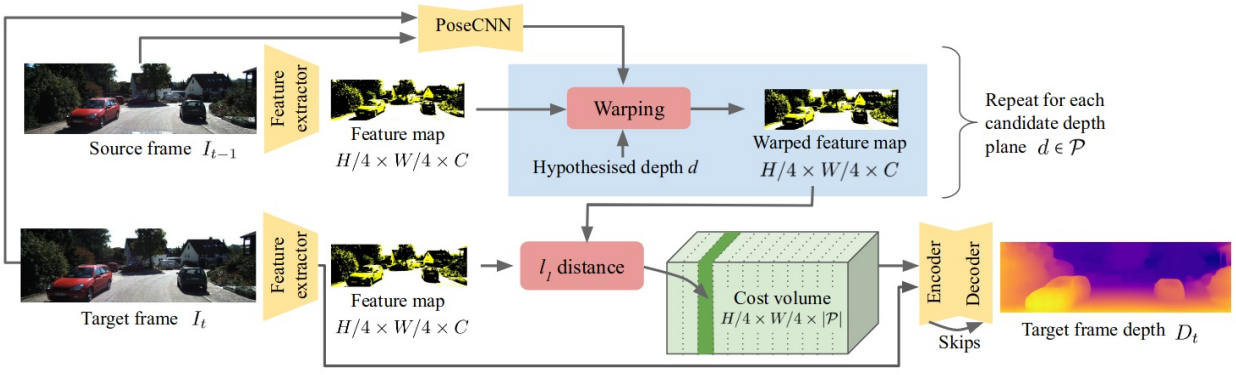
- Key idea: remove stationary pixels caused by static camera, objects moving at a similar velocity as camera

$$\mu = \left[\min_{t'} pe(I_t, I_{t' \to t}) < \min_{t'} pe(I_t, I_{t'}) \right]$$

Contribution from Manydepth

1. Adaptive cost volume

- Make cost volume be the input of the depth encoder (rather than the an image)
- Cost volume is a tensor with shape of num_bins * height * width (num_bins is the number of possible depths, e.g. we assume all pixels are among the depth from 0 m to 10 m with precision of 0.1 m, then the num_bins is (10-0)/0.1=100)
- Cost volume is basically telling us , for each pixel, what's the likelihood of the correct depth being at each candidate depth
- The computation of cost volume is similar to the reprojection loss above, except the depth is constant for all pixels
- Update the range of potential depth every training batch



Overview of cost volume in manydepth

2. Teacher network

- Cost volume based method is not reliable in region where objects are moving or surfaces are untextured
- Use a seperate non-cost-volume based method (monodepth2) to overwrite these areas
- In order to identify these unreliable pixels, the depth predicted by teacher network and the cost volume are utilized.

$$M = \max\left(\frac{D_{\text{cv}} - \hat{D}_t}{\hat{D}_t}, \frac{\hat{D}_t - D_{\text{cv}}}{D_{\text{cv}}}\right) > 1$$

Our works

1. Use case

Detect litters near the sweeper using vision method (rasing the redundancy of small objects detection system)

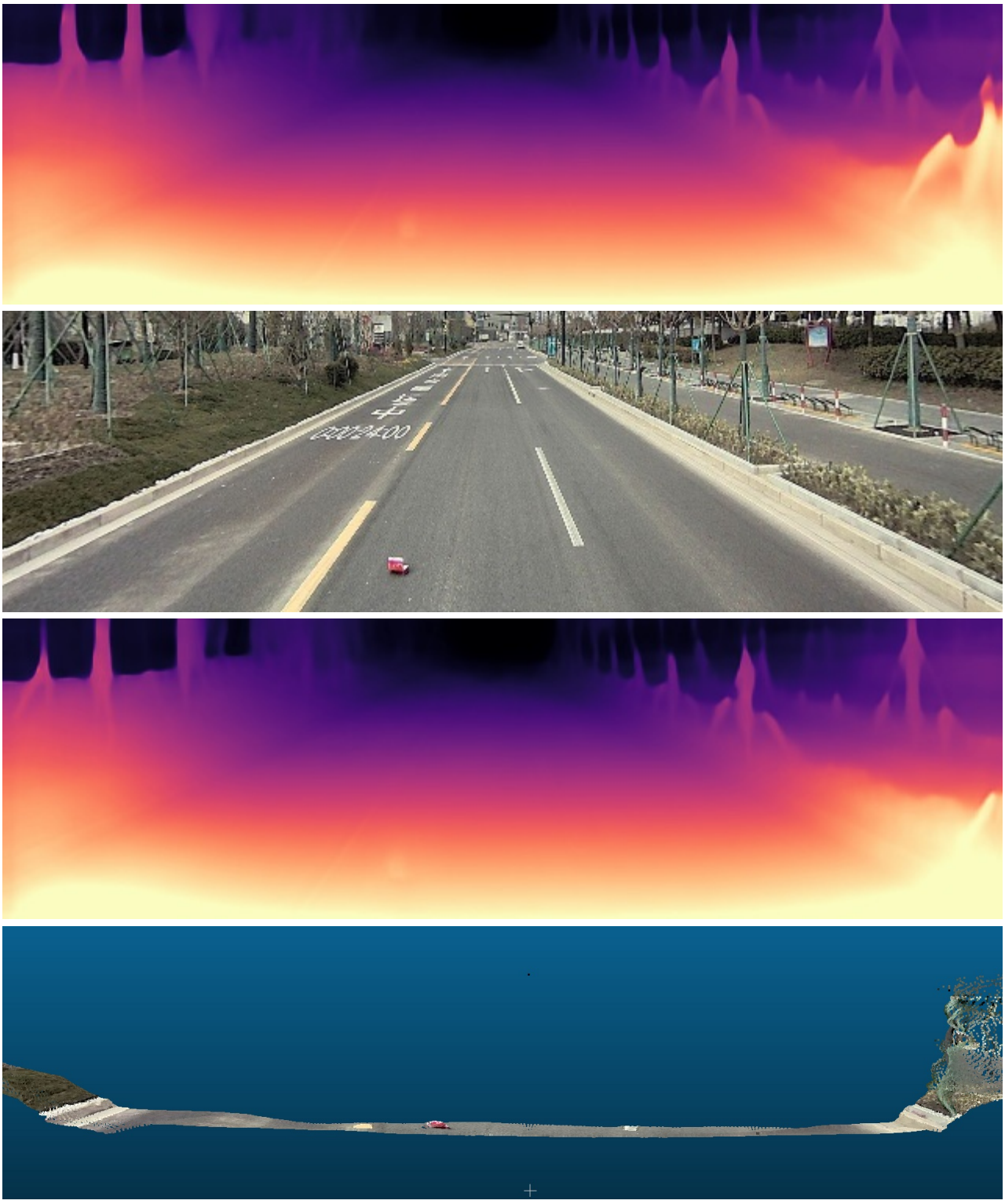
2. Pipeline

i. Extract and preprocess RGB images (lingang_map5/cam00, lingang_map5/cam04)
ii. Train Manydepth with autowise_dataset and cooresponding intrincis
iii. Inference RGB images into depth images
iv. Reproject depth image into 3D points in camera coordinate (dye using RGB image)
v. Concatanate isolated 3D points and transform them to a global coordinate with poses from nav_state

3. Progress

- The quality of estimated depth is not good enough to reconstruct reliable scene
- Initially, we want to evaluate the reconstructed point cloud with the dense point cloud generated by lidar. Since the reconstructed point cloud is terrible, it's better to evaluate the results with depth images
- Cropping sky is beneficial for GPU memory without affecting depth estimation in other areas
- The pixels with higher depth value are prone to be wrongly predicted and inconsistent with neighbor pixels. Use higher resolution images should alleviate this issue
- Data from wide-angle lens (cam00) perform better than that from telephoto lens (cam04) in qualitative results

4. Some results



rgb, estimated depth and reconstructed point cloud

Future works

- Write a evaluation module relying on depth images
- Collect more data with litters
- Use higher resolution images as inputs
- Multi card training setup (320*1664, batch_size=4, gpu memory=21664 MB)

- Use semantic mask to help depth_encoder and depth_decoder to maintain a depth consistency for the same object
- About the scale, in order to get a real world scale, we still need real poses or find a median scale during inference.
  - Depth map scale for KITTI data
  - Wrong depth scale when using ground-truth camera poses
- Data augmentation (Domain transfer)

Other related works

MonoRec (CVPR2021): Multi-step training scheme

Unsupervised High-Resolution Depth Learning from Videos with Dual Networks