Summary

1. Maps directly from input pixels to :
   - estimate of ego-motion (6-DOF transformation matrices) .
   - per-pixel depth map under a reference view.

2. The method is unsupervised and is trained with image sequences.

3. View synthesis is the task. to train the network.

4. They use single-view depth and multi-view pose networks, with a loss based on warping nearby views to the target using the computed depth and pose.

Unsupervised learning of depth and ego motion from video.

Loss function

1. The final loss function is :

$$\mathcal{L}_{final} = \sum_{l} \mathcal{L}_{vs}^{l} + \lambda_s \mathcal{L}_{smooth}^{l} + \lambda_e \sum_{s} \mathcal{L}_{reg}(\hat{E}_s^l).$$

Explanation of $\mathcal{L}_{vs}^{l}$:

1. $\mathcal{L}_{vs} = \sum_{s} \sum_{p} \left| I_t(p) - \hat{I}_s(p) \right|$ (without E explainability)

where p indexes over pixel coordinates and
s indexes over source view.

2. $\hat{I}_s$ is the source view I_s warped to the target coordinate frame based on a depth image-based rendering module.

3. Prev methods require the pose while they predicts the pose as part of the learning framework.

4. The depth image-based rendering takes as input :

- predicted depth $\hat{D}_t$
- predicted 4×4 camera transformation matrix $\hat{T}_{t \to s}$
- source view I_s

5. $p_s \sim K \hat{T}_{t \to s} \hat{D}_t(p_t) K^{-1} p_t$

homog. coordinates of a pixel in the target view

camera intrinsics

predicted depth map.

predicted relative pose.

6. They additionally train an explanability prediction network that outputs a per-pixel soft mask $\hat{E}_s$ for each target-source pair.

7. So $\mathcal{L}_{vs} = \sum_s \sum_p \hat{E}_s(p) \left| I_t(p) - \hat{I}_s(p) \right|$

8. $\hat{E}$ supposed to indicate where view synthesis can be expected to be meaningful.