Scene Semantic Reconstruction from Egocentric RGB-D-Thermal Videos
-
Supplementary Materials

Rachel Luo, Ozan Sener, and Silvio Savarese
Stanford University
{rsluo, osener, ssilvio}@stanford.edu

Abstract

In this Supplementary Materials section, we provide implementation details missing in the main paper. We also provide a video summary and several qualitative results in the attached supplementary video.

1. Details of ORB-SLAM

We extend ORB-SLAM [3] to create an egocentric SLAM algorithm that can combine the three data modalities (RGB, depth, and thermal) and can handle both static and dynamic objects. A description of ORB-SLAM is below, and we refer the readers to [3] for the full details and implementation.

ORB-SLAM is a SLAM system for monocular, stereo, or RGB-D cameras, and it works in real-time in a wide variety of static environments. It runs three parallel threads: i) tracking to localize the camera at each frame; ii) local mapping and bundle adjustment; and iii) loop closure to correct any accumulated drift. The system uses the same ORB features as in [2], and it pre-processes the input to extract features at salient keypoint locations. It uses motion-only bundle adjustment to optimize the camera pose in the first thread, local bundle adjustment to optimize the local window of keyframes and points in the second thread, and full bundle adjustment after a loop closure to to optimize all keyframes and points in the third thread. ORB-SLAM yields especially accurate and robust results for camera localization.

2. Details of the Semantic Segmentation

Our semantic segmentation pipeline closely follows [5] in terms of energy minimization. In this section, we first provide a detailed explanation of the energy function definition and then discuss the optimization process. Our semantic segmentation pipeline is composed of two steps: i) segmenting the hands from the scene; and ii) segmenting the dynamic object from the remaining part of the scene.

For hand segmentation, we use the output of the hand detector as a prior, along with the fact that the hand is typically warm compared to its surroundings. We also keep a hand color model in terms of a Gaussian Mixture Model (GMM) throughout the segmentation. Our algorithm runs in an online fashion as required in SLAM; hence, future segmentations cannot affect the previous frames. Our energy function is defined as

$$\min_{\alpha_i^t} \sum_i U(\alpha_i^t, \mathbf{y}_i^t) + \sum_i \sum_{j \in \mathcal{N}(i)} V(\mathbf{y}_i^t, \mathbf{y}_j^t) 1[\alpha_i^t \neq \alpha_j^t] \quad (1)$$

In this formulation, α_i^t is a binary variable which is 1 if pixel i is part of the hand at time t and 0 otherwise. $\mathcal{N}(i)$ is the set of pixels neighboring i, and $1(\cdot)$ is an indicator function. $\mathbf{y}$ is the concatenated vector of $\mathbf{z}, \mathbf{c}, d, \tau$. We also normalize position, color, and temperature values to $[0, 1]$.

$U(\alpha_i^t, \mathbf{y}_i^t)$ is the unary energy representing the likelihood that the i^{th} pixel is part of the hand. It is a weighted combination of the likelihood over the temperature (T), color (C), hand-detector outputs (S), and history over time (H).

$$\begin{aligned} U(\alpha_i^t, \mathbf{y}_i^t) &= w_T U^T(\alpha_i^t, \mathbf{y}_i^t) + w_C U^C(\alpha_i^t, \mathbf{y}_i^t) \\ &+ w_S U^S(\alpha_i^t, \mathbf{y}_i^t) + w_H \sum_i U(\alpha_i^{t-1}, \mathbf{y}_i^{t-1}) e^{-\Delta(\mathbf{y}_i^t, \mathbf{y}_i^{t-1})} \end{aligned} \quad (2)$$

where $\Delta(\cdot, \cdot)$ is the geodesic distance over all modalities between two points, defined as the minimum sum of color, depth, and temperature differences along a path between two points. $V(\cdot, \cdot)$ is the binary consistency term defined over neighboring pixels as

$$V(\mathbf{y}_i^t, \mathbf{y}_j^t) = \exp\left(-\frac{|\mathbf{y}_i^t - \mathbf{y}_j^t|_2}{\gamma}\right) \quad (3)$$

where $\gamma = \frac{1}{N} \sum_i \frac{1}{|\mathcal{N}(i)|} \sum_{j \in \mathcal{N}(i)} |\mathbf{y}_i^t - \mathbf{y}_j^t|_2$, and N is the total number of pixels.

We define the each component of the unary energy as

$$\begin{aligned} U^T(\alpha_i^t, \mathbf{y}_i^t) &= \tau_i^t 1[\alpha_i^t = 1] + (1 - \tau_i^t) 1[\alpha_i^t = 0] \\ U^C(\alpha_i^t, \mathbf{y}_i^t) &= p(\mathbf{c}_i^t | \alpha_i^t) 1[\alpha_i^t = 1] \\ U^S(\alpha_i^t, z_i^t) &= \left(\sum_{k \in \mathcal{H}} p_k e^{-\Delta(\mathbf{y}_i^t, \mathbf{y}_k)} \right) 1[\alpha_i^t = 1] \end{aligned} \tag{4}$$

where $p(\mathbf{c}_i^t|\alpha_i^t)$ is an RGB-color model represented as a GMM with five components and learned separately for the hand and the static scene. $\mathcal{H}$ is a collection of hand detections, each represented by a centroid $\mathbf{c}_k$ and a detection likelihood p_k. $\mathbf{y}_k$ is the color, position, depth, and temperature of the centroid of the detected hand. Here, $U^C(\alpha_i^t, \mathbf{y}_i^t)$ and $U^S(\alpha_i^t, z_i^t)$ are only defined for positive variables. To make this work as a binary energy minimization problem, we normalize each term by dividing it by its maximum value; we then use $1 - U^C(\alpha_i^t, \mathbf{y}_i^t)$ and $1 - U^S(\alpha_i^t, z_i^t)$ for the background likelihood.

All components of this energy function can be computed in log-linear time using bi-linear filters, and minimized using the min-cut/max-flow framework as explained in [5].

After the hands are segmented, we segment the rest of the image into static and dynamic object components. We use the same energy minimization framework after introducing an additional prior on motion and dropping the prior on color. The motion prior corresponds to the fact that the motion of the object in interaction is different from the camera motion and is defined as:

$$U^M(\alpha_i^t, \mathbf{y}_i^t) = \rho(|\mathbf{z}_i^t - \mathbf{z}^{t-1}_{\pi(\mathbf{R}^t \mathbf{x}_i^t + \mathbf{t}^t)}|) 1[\alpha_i = 0] \tag{5}$$

where ρ is the Huber function, π is the pinhole projection, $\mathbf{R}, \mathbf{t}$ are the estimated camera pose (rotation and translation, respectively), and $\mathbf{X}_i$ is the 3D position of i^{th} point in homogeneous coordinates. With some abuse of notation, α_i is a binary variable which is 1 if pixel i is part of the object in interaction and 0 otherwise. Since this function is only defined for the likelihood of being the object in interaction, we again compute the background likelihood as $1 - U^M(\alpha_i^t, \mathbf{y}_i^t)$ after normalizing it to $[0, 1]$. Please note that (with abuse of notation) we used α_i^t to denote a point that is part of the dynamic object vs. the static background in this energy minimization framework. Hence, the final energy minimization is over

$$\begin{aligned} U(\alpha_i^t, \mathbf{y}_i^t) &= w_T U^T(\alpha_i^t, \mathbf{y}_i^t) + w_M U^M(\alpha_i^t, \mathbf{y}_i^t) \\ &+ w_S U^S(\alpha_i^t, \mathbf{y}_i^t) + w_H \sum_i U(\alpha_i^{t-1}, \mathbf{y}_i^{t-1}) e^{-\Delta(\mathbf{y}_i^t, \mathbf{y}_i^{t-1})} \end{aligned} \tag{6}$$

and $V(\mathbf{y}_i^t, \mathbf{y}_j^t)$.

Since we have annotations on segmentation for a subset of videos, we used these annotated videos for cross-validation to choose parameters w_T, w_S, w_H, w_M.

3. Details of the Multi-Modal Distillation

Intuitively, both the depth and the thermal modalities should be useful for hand detection. However, our dataset is rather small for training from scratch and thus, we would need an existing dataset for pre-training (along the lines of ImageNet) that includes depth and thermal modalities. Since no such pre-trained models exist, we instead use the recently proposed knowledge distillation method [1]. We use pre-training software provided by the authors of [4] and perform knowledge distillation using the l_2 difference at the representation layer (i.e. the fully connected layer before the detections). We distill knowledge from the network pre-trained on ImageNet and distributed by [4].

References

[1] S. Gupta, J. Hoffman, and J. Malik. Cross modal distillation for supervision transfer. In *In Proc. Computer Vision and Pattern Recognition (CVPR)*, 2016.

[2] M. J. M. M. Mur-Artal, Raúl and J. D. Tardós. ORB-SLAM: a versatile and accurate monocular SLAM system. *IEEE Transactions on Robotics*, 31(5):1147–1163, 2015.

[3] R. Mur-Artal and J. D. Tardós. ORB-SLAM2: an open-source SLAM system for monocular, stereo and RGB-D cameras. *arXiv preprint arXiv:1610.06475*, 2016.

[4] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 779–788, 2016.

[5] O. Sener, K. Ugur, and A. A. Alatan. Efficient mrf energy propagation for video segmentation via bilateral filters. *IEEE Transactions on Multimedia*, 16(5):1292–1302, Aug 2014.