Label Propagation in RGB-D Video

Md. Alimoor Reza, Hui Zheng, Georgios Georgakis, Jana Košecká

Abstract—We propose a new method for the propagation of semantic labels in RGB-D video of indoor scenes given a set of ground truth keyframes. Manual labeling of all pixels in every frame of a video sequence is labor intensive and costly, yet required for training and testing of semantic segmentation methods. The availability of video enables propagation of labels between the frames for obtaining a large amounts of annotated pixels. While previous methods commonly used optical flow motion cues for label propagation, we present a novel approach using the camera poses and 3D point clouds for propagating the labels in superpixels computed on the unannotated frames of the sequence. The propagation task is formulated as an energy minimization problem in a Conditional Random Field (CRF). We performed experiments on 8 video sequences from SUN3D dataset [1] and showed superior performance to an optical flow based label propagation approach. Furthermore, we demonstrated that the propagated labels can be used to learn better models using data hungry deep convolutional neural network (DCNN) based approaches for the task of semantic segmentation. The approach demonstrates an increase in performance when the ground truth keyframes are combined with the propagated labels during training.

I. INTRODUCTION

Semantic segmentation is one of the ingredients of scene understanding beneficial to a variety of robotic tasks. For example capability of semantic parsing on indoors scenes supports better localization, context understanding for recognition and or manipulation or path planning and navigation. Semantic segmentation requires simultaneous segmentation and categorization of image regions and assigning semantic category labels to image pixels. Most effective machine learning approaches for this task use either Deep Convolutional Neural Networks (DCNN) or Conditional Random Fields (CRF) for this task. Training of these models requires pixel level ground truth labels and the annotation process is often costly and labor intensive. In case of video sequences, several previous works explored the strategy of annotating few keyframes and used label propagation techniques to obtain labels in additional frames. Some of the representative methods were initially developed for video sequences with multiple moving objects and static backgrounds.

In this work, we describe a novel label propagation method for indoor RGB-D video sequences captured by either handheld camera or camera mounted on robotic platform. Our method overcomes optical-flow challenges of low textured indoors scenes using superpixel representations of indoors scene along with its dominant 3D planes estimated from 3D point cloud and exploits the ego-motion between the frames for registration and general camera pose estimation. We propose an energy based formulation of the label propagation exploiting superpixels and evaluate the proposed method on a subset of withheld ground truth frames.

We further investigate the effectiveness of using the propagated labels for training Deep Convolutional Neural Network (DCNN) model for the task of semantic segmentation. Experimenting with various combination of manually annotated labels and propagated labels we concluded that the propagated labels are useful to learn models that give improved performance for the semantic segmentation. In summary, the contributions of the paper are listed as follows:

i) We introduce novel energy minimization formulation of label propagation for RGB-D videos of indoors scenes utilizing camera poses and 3D point clouds aggregated...
through superpixels. The choice of superpixels enables us to deal favorably with large untextured regions often present in indoors scenes. The use of camera pose and 3D geometry for propagation overcomes some of the difficulties of traditionally used optical flow based methods [4], [5].

ii) We evaluate experimentally the effect of using different proportions of the propagated labels for training a Deep Convolutional Neural Network (DCNN) [2] for the task of semantic segmentation demonstrating that the additional training data improves the performance. The evaluation is carried out on RGB-D videos of SUN3D [1] scenes. An overview of our approach is shown in Figure 1.

II. RELATED WORK

Here we briefly review the works related to ours in label propagation and representative works in the areas of semantic segmentation and generation of additional training data for deep CNNs. The previous approaches for dense label propagation in videos varied in the types of scenes considered or types of ground truth annotations.

Label Propagation: Badrinarayanan et al. [6] proposed an HMM-based inference method for transferring labels from the first and last annotated keyframes into the remaining unlabeled frames for outdoor video sequences. The work of [4] suggested a probabilistic formulation of the problem by combining information from optical flow, appearance and spatial proximity cues from adjacent labeled frames to the unlabeled ones. Authors in [5] more recently followed a similar approach for label propagation in outdoor video frames and demonstrated the efficacy of the propagated labels for learning a better model for semantic segmentation. Miksik et al. [7] introduced a filtering algorithm that predicts per-pixel label distribution from a separate model in the current frame, then it temporally smooths out the prediction from previous frame. All of these approaches use forward or backward optical flow computation. The flow has difficulties in the presence of large motions, textureless regions and occlusions, which are all abundant in indoors environments. Inaccurate flow can cause errors in the labeling and is unable to handle disappearance and reappearance of objects in the FOV.

Semantic Segmentation: The task of semantic segmentation has been tackled by strategies using multi-class CRFs or classification of bottom-up segmentation using hand-engineered features [8], [3], [9], [10]. Recent adoption of Deep Convolutional Neural Network (DCNN) for semantic segmentation surpassed the earlier approaches in predictive performance. The DCNN based approaches leverage effective feature learning and end-to-end pixel level training. One pioneering example of such strategies is the fully convolutional network (FCN) [11]. Building on the success of FCN, the other DCNN approaches refine the fully convolutional output with the help of CRFs [12] or other global energy models [13] to get better boundaries of the predicted region.

Synthetic Data: There are several works that address the problem of the limited availability of training data by generating synthetic data for a specific task [14], [15], [16], [14] used 3D CAD object models to generate images by randomizing the pose of the objects for the purpose of training object detectors. For semantic segmentation, the work of [15] leverages the Grand Theft Auto video game engine in order to generate per-pixel annotations for realistic scenes in the game, while [16] takes advantage of 3D synthetic scenes generated from CAD models to create annotations from any arbitrary pose. All of these approaches, however, create annotations using synthetic computer graphics generated data. Models trained on synthetic data alone typically underperform when applied on real images, therefore, the synthetic data are usually used to augment existing real training sets in order to boost the performance. In contrast, label propagation approaches produce annotations for real images by taking advantage of pre-existing but limited number of annotations. The expected performance improvement however is not as large, since the additional training examples are previously included categories, with slightly bigger viewpoint variations.

III. APPROACH

To overcome difficulties of the pixel label propagation techniques, we formulate the propagation method on superpixels. For each unlabeled frame, we use the two nearest labeled keyframes to propagate label information via associated 3D point cloud. The ground truth image labels are first projected to 3D point cloud and using the camera pose information transformed to the unlabeled frame to provide label evidence for each superpixel. Finding the most likely label for all the pixels in an unlabeled frame is formulated as energy minimization in Conditional Random Field (CRF) framework. Experimentally we show that the propagated labels are useful for training better Deep Convolutional
we wish to minimize the following energy function:

\[ E(X_k|I_k, A_l, A_{l+1}) = \sum_{i \in V} \theta_i(x_i; I_k, A_l, A_{l+1}) + \sum_{(i,j) \in \zeta} \psi_{ij}(x_i, x_j; I_k, A_l, A_{l+1}) \]  

(1)

Here \( \theta_i(\cdot) \) and \( \psi_{ij}(\cdot) \) are the unary and pairwise energy functions respectively. The CRF graph \( G = (V, \zeta) \) is defined over the pixels in the image \( I_k \) and we follow a 4-connected neighborhood system. We utilize the two closest labeled keyframes from \( I_k \) namely \( A_l \) and \( A_{l+1} \) from the available set of labeled keyframes. Let’s assume \( I_m \) and \( I_n \) stand for the images corresponding to labeled frames \( A_l \) and \( A_{l+1} \) respectively. Then the frame \( I_k \) which is subjected to propagation in Equation 1 lies in between the \( I_m \) and \( I_n \) such that \( m < k < n \). We encode the unary and pairwise energy terms as follows:

**Unary Term:** From the labeled keyframes \( A_l \) and \( A_{l+1} \), we get the labeled 3D point cloud using the camera pose information and project it into each superpixel. Within the superpixel, we distribute the same score to all the pixels. Our unary term is computed as follows:

\[ \theta_i(x_i; I_k, A_l, A_{l+1}) = -F(x_i; I_k, A_l, A_{l+1}) \]  

(2)

where \( F(\cdot) \) is the scoring function for the superpixel that encompasses the pixel \( i \). This function estimates the probability of semantic labels \( \{c_1, c_2, ..., c_L\} \). Using the camera poses (rotation, translations), namely \((R_l, T_l)\) and \((R_{l+1}, T_{l+1})\), we project the labeled point clouds into the current frame \( I_k \) using the standard camera perspective projection. For each superpixel, we count the number of projected pixels with a particular label \( c_j \). Then we find the ratio of this count with the size of the superpixel as the score \( f^j_k \) for label \( c_j \). We can get a similar score \( f^j_{l+1} \) from the other labeled frame. These two scores are averaged to give us the final score \( F(\cdot) \) for label \( c_j \). This superpixel score is distributed across all the pixels inside it to give us pixel level scores on which we optimize the energy function in Equation 1.
**Fig. 4:** Qualitative results for the label propagation experiment on the 8 video sequences from SUN3D [1]. From left to right we show the RGB image, the ground truth (GT), the optical-flow-based label propagation (OF), and the 3D point cloud projection-based label propagation (Ours).

**Pairwise Term:** The pairwise energy function is enforced by a simple Potts model, which penalizes the adjacent pixels with different labels as follows:

$$
\psi_{ij}(x_i, x_j; I_k, A_l, A_{l+1}) = \begin{cases} 
0, & l_i = l_j \\
b, & l_i \neq l_j 
\end{cases}
$$

where $b$ has been empirically set to 2.5 for all our experiments. This enforces a smoothness in the label prediction. The proposed energy is then minimized using Graph-Cuts [18]. Our label propagation algorithm is depicted in Figure 3.

**Table I:** Statistics of the 8 video sequences in SUN3D [1]. KF-all shows the total number of keyframes per scene, KF-prop refers to the number of keyframes used to propagate the labels into the unannotated frames, and KF-eval shows the number of keyframes used for evaluation of our label propagation approach.

**Fig. 5:** Comparison between the baseline optical-flow-based (middle row) and our projection-based (bottom row) label propagation in a video sequence. Notice that pillows get out of the field-of-view on frame 39 and OF fails to recover their label in frames 51 and 64 when they come into view again. In contrast, our approach has no problem retrieving the correct labels. Best viewed in color.

**Fig. 6:** Semantic segmentation results on scenes from the SUN3D [1] dataset. From left to right we show the RGB image, the manually-annotated ground truth, and the results from the models trained with the GT, GT+Prop-small, and GT+Prop-large. When using only the keyframes to train (GT) we notice that the segmentation is often cluttered with wrong label predictions. In contrast, the models trained with a combination of keyframes and propagated labels GT+Prop-small, and GT+Prop-large produce a smoother output with clearer boundaries for the semantic classes. Note also that for some images, a large portion of the ground truth annotations are missing (rows 1 and 3) for which the semantic segmentation predicts the correct semantic classes.

**IV. Experiments**

We validated our approach on the 8 RGB-D video sequences from SUN3D [1]. In each video few keyframes are annotated using the LabelMe [19]. We use a subset of these keyframes to propagate labels into the unlabeled ones and the rest of the keyframes to validate our propagation in a video sequence. Table I shows the statistics for the video sequences. We used 11 object classes: **Unknown, Bed, Ceiling, Chair, Floor, Furniture, Objects, Picture, Table, Wall, and Window.** We measure the performance using three different metrics i) **average per-class:** proportion of correctly labeled pixels for each class then average these proportions, ii) **average IoU:** finds the intersection over union of the labeled segments for each class then computes the averages over classes, and
TABLE II: Label propagation accuracies (%) in different metrics-average per class, average IoU, and global-on the SUN3D video sequences [1]. We compare to the optical-flow-based (OF) baseline and demonstrate superior performance.

<table>
<thead>
<tr>
<th></th>
<th>Unknown</th>
<th>bed</th>
<th>ceiling</th>
<th>chair</th>
<th>floor</th>
<th>furniture</th>
<th>objects</th>
<th>picture</th>
<th>room</th>
<th>wall</th>
<th>window</th>
<th>mean</th>
<th>Global</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Average per class</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OF</td>
<td>42.3</td>
<td>96.5</td>
<td>54.7</td>
<td>83.3</td>
<td>89.4</td>
<td>92.6</td>
<td>83.6</td>
<td>88.7</td>
<td>94.8</td>
<td>90.6</td>
<td>92.3</td>
<td>82.6</td>
<td>80.3</td>
</tr>
<tr>
<td>Ours</td>
<td>57.0</td>
<td>98.8</td>
<td>69.3</td>
<td>90.0</td>
<td>93.3</td>
<td>96.1</td>
<td>88.8</td>
<td>91.5</td>
<td>97.6</td>
<td>97.5</td>
<td>97.5</td>
<td>88.0</td>
<td>87.4</td>
</tr>
<tr>
<td><strong>Average IoU</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OF</td>
<td>35.7</td>
<td>93.9</td>
<td>52.7</td>
<td>71.5</td>
<td>67.0</td>
<td>83.3</td>
<td>68.4</td>
<td>65.5</td>
<td>81.9</td>
<td>75.9</td>
<td>84.4</td>
<td>72.8</td>
<td></td>
</tr>
<tr>
<td>Ours</td>
<td>50.2</td>
<td>98.1</td>
<td>66.2</td>
<td>78.9</td>
<td>76.2</td>
<td>89.8</td>
<td>83.4</td>
<td>89.5</td>
<td>89.7</td>
<td>87.2</td>
<td>80.3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

B) to the optical-flow-based baseline in the average per class, average IoU, and global accuracy metrics and show an average improvement of 5.4%, 7.5% and 7.1% respectively over the baseline. Figure 4 presents some qualitative examples where we observe the superior performance of our approach. For instance, in the bottom row, optical-flow completely missed the categories of Floor, Table, and most of Chair. This is clearly illustrated in Figure 5 where optical flow fails to recover a label for an object that went out of view for a certain number of frames.

ii) **Global** accuracy: proportion of correctly labeled pixels for all classes together. Next we describe the results of our experimentation.

A. Label Propagation

In each video sequence, we used 75% of the total keyframes to propagate labels into the unannotated frames and the remaining 25% to validate the propagation. Table I column 4 shows the number of keyframes used for propagation. We used 93 in total keyframes during evaluation (in Table I last column). We compared our results against a propagation scheme, which uses similar CRF formulation to ours; instead of using the 3D labeled point cloud projection score, we utilized optical flow to transfer labels from an adjacent frame. This method is similar to the methods of [5], [4] that use optical flow and appearance cues for label propagation. We refer to this optical-flow-based propagation as our baseline.

Optical Flow Based Label Propagation: The baseline optical-flow-based label propagation differs from our projection-based label propagation in the way we compute the unary term in the CRF energy in Equation 1. In this approach, the labels are always propagated from the previous frame. Optical-Flow\(^1\) is computed from frame \(I_k\) to frame \(I_{k-1}\). Having the optical flow computed for each pixel in the current frame \(I_k\), we can accumulate the optical flow vectors inside a superpixel \(s^k_l\) to find a set of superpixels \(S_{k-1} = \{s^k_{l-1}, s^k_{l-2}, ..., s^k_{l-k}\}\) in the previous frame \(I_{k-1}\). This set of superpixels \(S_{k-1}\) in the previous frame indicates where the collective optical flows of the pixels inside superpixel \(s^k_l\) will lead it. We find the color-based appearance similarities between superpixels in the set \(S_{k-1}\) and superpixel \(s^k_l\). This similarity is weighted by a size based score. More specifically, the scoring function \(G(\cdot)\) of unary term for label \(l\) is computed as follows:

\[
G(s^k_l; I_{k-1}) = \phi(s^k_l, s^k_{l-1}) IoU(s^k_l, s^k_{l-1})
\]

Here \(\phi(s^k_l, s^k_{l-1})\) is \(\chi^2\) distance between the HSV color histograms. \(IoU(s^k_l, s^k_{l-1})\) is intersection over union ratio between the two superpixels.

\(^1\)We used the open source implementation of the Matlab toolbox https://www.cs.cmu.edu/~katef/LDOF.html

Results: In Table II we present the evaluation of our label propagation algorithm on 11 object categories. We compare our projection-based propagation (described in III-B) to the optical-flow-based baseline in the average per-class, average IoU, and global accuracy metrics and show an average improvement of 5.4%, 7.5% and 7.1% respectively over the baseline. Figure 4 presents some qualitative examples where we observe the superior performance of our approach. For instance, in the bottom row, optical-flow completely missed the categories of Floor, Table, and most of Chair. This is clearly illustrated in Figure 5 where optical flow fails to recover a label for an object that went out of view for a certain number of frames.

B. Semantic Segmentation with Propagated Labels

Next we show the effect of using additional labels for learning DCNN models for semantic segmentation task. Many DCNN architectures have been proposed for semantic segmentation e.g., Fully Convolutional Network (FCN) [11], DeepLab [12], SegNet [2] to mention a few. To show the effectiveness of the propagated labels, we selected the Encoder-Decoder architecture of SegNet [2] and trained it for semantic segmentation of ten object classes Bed, Ceiling, Chair, Floor, Furniture, Objects, Picture, Table, Wall, and Window in indoor scene. Following up with our label propagation experiment, we partitioned the 8 video sequences of SUN3D [1] into equal halves of 4 training and 4 testing video sequences. The 4 training videos have 264 keyframes (video sequences hotel-umd, hv-c5, studyroom, and mit-32 in Table I). We used the 199 keyframes to train our baseline semantic segmentation model (video sequences hotel-umd, hv-c5, studyroom, mit-32 in Table I column 4). This model is referred to as GT. Additionally, we prepared two subsets of training sets with propagated labels of different sizes; Prop-small and Prop-large. For Prop-small, we maintained a similar size (199 images in total) to our baseline GT of annotated keyframes. Let us define a propagation interval to be the set of unannotated frames in between two consecutive keyframes from which labels are propagated (see III-B). We randomly sample a propagated frame from all the frames within a propagation interval. Accumulating all such samples across all the propagation intervals from the training video sequences make up the training set Prop-small. Prop-large consists of a set of 2488 images with propagated labels by taking every 5-th frame from each of the training video
sequences. Additionally, we also created two other training sets by augmenting the manually-annotated keyframes with our two sets of training images with propagated labels. These two different sets are referred to as GT+Prop-small and GT+Prop-large. All the trained models are evaluated on all the 119 keyframes of 4 test videos (dorm, hv-c6, hv-c8, mil-lab in Table I in column 3). For the training sets of GT, Prop-small, GT+Prop-small we trained the models for 40000 iterations, and for the sets of Prop-large and GT+Prop-large we trained for 50000 iterations. For all the models we used a learning rate of 0.001. SegNet is trained using the cross-entropy loss function. In order to mitigate the dominance of more frequently appearing classes over the less frequent ones, we followed the median-frequency-balancing scheme of [20] to compute the weights associated with each class as suggested by the authors of SegNet [2].

Results: The results of our semantic segmentation with propagated labels are tabulated in Table III. The performance is reported in the metrics of global, average per-class, and average IoU accuracy, as shown in [2]. We observe that by slightly augmenting the current keyframes (GT+Prop-small) we get an increase in performance of 3% in global accuracy, 4.5% in average IoU, and 0.4% in average per-class accuracy. The best performing model is GT+Prop-large which attains a 5.7% increase in global accuracy, a 5.3% increase in average IoU, and a 2.2% increase in average per-class accuracy compared to the GT training set. These results suggest the effectiveness of our propagated labels for learning better models for semantic segmentation. We show some qualitative comparisons of the predicted images from these different models in Figure 6.

In the case of Prop-large, which uses only the propagated labels for training, we get improvement in global, average IoU accuracies, but we observe a small drop in the average per class compare to GT. When we use Prop-small which has the same size as GT, we notice a slight degradation of performance in all three metrics. This is expected due to the fact that small errors in labeling are introduced during the propagation stage. One such error is the inconsistency in labeling a same region in multiple keyframes.

V. CONCLUSIONS

We presented an approach for label propagation in RGB-D video sequences which introduces a novel energy minimization formulation in a Conditional Random Field (CRF). The label propagation is facilitated by taking advantage of the camera poses of the frames and 3D point clouds. Our experiments reveal that our approach outperforms an optical flow based propagation in different evaluation metrics. Furthermore, we have demonstrated that the propagated labels can be effectively utilized for training DCNNs for semantic segmentation. Models trained with our propagated labels perform comparably and sometimes better than the models trained with the manually-annotated keyframes only. This suggests that the propagated labels are consistent throughout the video sequence and provide with accurate annotations.

TABLE III: Semantic segmentation accuracies comparison in three different metrics (%): Global, Average Per-class, and average IoU.

<table>
<thead>
<tr>
<th>Train set</th>
<th>Global</th>
<th>Average Per-Class</th>
<th>average IoU</th>
</tr>
</thead>
<tbody>
<tr>
<td>GT</td>
<td>65.1</td>
<td>50.3</td>
<td>36.4</td>
</tr>
<tr>
<td>Prop-small</td>
<td>64.0</td>
<td>46.2</td>
<td>34.1</td>
</tr>
<tr>
<td>Prop-large</td>
<td>70.1</td>
<td>49.8</td>
<td>39.4</td>
</tr>
<tr>
<td>GT+Prop-small</td>
<td>68.1</td>
<td>50.7</td>
<td>40.9</td>
</tr>
<tr>
<td>GT+Prop-large</td>
<td>70.8</td>
<td>52.5</td>
<td>41.7</td>
</tr>
</tbody>
</table>

REFERENCES