Finding Temporally Consistent Occlusion Boundaries in Videos using Geometric Context

S. Hussain Raza
Nvidia Corporation
Santa Clara, CA, USA.

Ahmad Humayun Irfan Essa
College of Interactive Computing
Georgia Institute of Technology, Atlanta, GA, USA.

Matthias Grundmann
Google Research
Mountain View, CA, USA.

David Anderson
School of Electrical and Computer Engineering
Georgia Institute of Technology, Atlanta, GA, USA.

http://www.cc.gatech.edu/cpl/projects/temporaloccl

Abstract

We present an algorithm for finding temporally consistent occlusion boundaries in videos to support segmentation of dynamic scenes. We learn occlusion boundaries in a pairwise Markov random field (MRF) framework. We first estimate the probability of an spatio-temporal edge being an occlusion boundary by using appearance, flow, and geometric features. Next, we enforce occlusion boundary continuity in a MRF model by learning pairwise occlusion probabilities using a random forest. Then, we temporally smooth boundaries to remove temporal inconsistencies in occlusion boundary estimation. Our proposed framework provides an efficient approach for finding temporally consistent occlusion boundaries in video by utilizing causality, redundancy in videos, and semantic layout of the scene. We have developed a dataset with fully annotated ground-truth occlusion boundaries of over 30 *videos (*$\sim$*5000 frames). This dataset is used to evaluate temporal occlusion boundaries and provides a much needed baseline for future studies. We perform experiments to demonstrate the role of scene layout, and temporal information for occlusion reasoning in dynamic scenes.*

1. Introduction

Objects in a scene exhibit occlusion due to their depth ordering with respect to the camera. In video, occlusion relationships can change over time due to ego-motion or movement of the objects themselves. In both cases, edges of the objects give occlusion boundaries. These occlusion boundaries are a strong indicator of object segmentations. Hoiem *et al.*[11] showed that by reasoning about occlusions, object segmentation, recognition, and scene description in images can be improved. To locate these edges, some initial estimates of motion and segmentations are required, but typical algorithms tend to fail close to these boundaries due to depth inconsistency. In this paper, we estimate these occlusion boundaries by combining low level appearance and flow cues with higher level information like geometric scene labels. These estimates of boundaries provide significant improvements to spatio-temporal video segmentation.

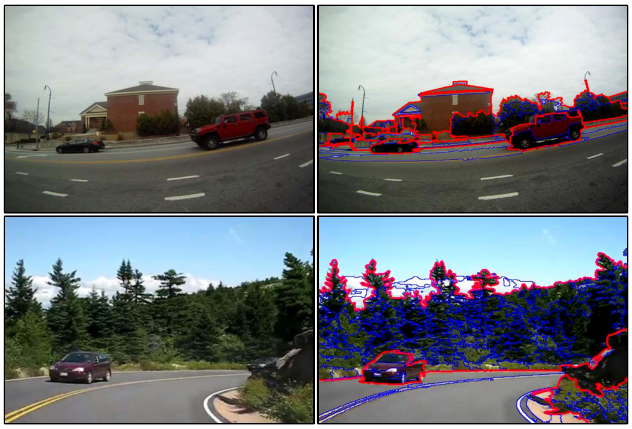

Figure 1. Video frames of an urban scene, occlusion and non-occlusion boundaries are labeled as red and blue, respectively. We demonstrate importance of geometric features and temporal redundancy for finding temporally consistent occlusion boundaries.

Our algorithm learns temporally consistent occlusion boundaries in dynamic scenes by leveraging spatio-temporal segmentation of videos. We first segment a video into spatio-temporal super-voxels [8, 26]. Over-segmentation provides a large number of candidate boundaries for learning occlusion/non-occlusion boundaries. We extract a broad range of features from each segment's

978-1-4799-6683-7/15 $31.00 © 2015 IEEE
DOI 10.1109/WACV.2015.141

IEEE computer society

boundary, and train unary and pairwise boundary classifier and enforce occlusion boundary continuity in MRF. MRF enables us to encode pairwise edgelet relations into our model, *i.e.*, probability of an occlusion boundary to be connected to other occlusion and non-occlusion boundaries, reducing false positives. We also demonstrate that aggregating information about occlusion boundaries over a temporal window increases performance when compared to a frame by frame approach. For testing and evaluations, we have developed a large dataset consisting of outdoor videos, annotated with occlusion boundaries.

Our primary contributions are: (1) a method for estimating temporally consistent occlusion boundaries by combining appearance, flow, and semantic scene information in an MRF framework; (2) a thorough evaluation of our algorithm by examining feature importance in estimating occlusion boundaries and comparison with other occlusion boundary algorithms (see Section 5); (3) in addition, we introduce a novel dataset of 30 annotated videos ($\sim$5,000 frames) with temporal occlusion boundaries and semantic information, as existing datasets do not provide temporal and semantic annotations.

2. Related Research

Geometric layout and temporal consistency in a dynamic scene provide strong cues for scene understanding and object segmentation. Hoiem *et al.*[11] demonstrated importance of geometric features for occlusion detection for images. Saxena *et al.*[20] proposed a planar model for estimating 3D structure of the scene from a single image. Applying image-based methods to individual video frames can provide occlusion reasoning of the dynamic scene. However, such image-based methods may not exploit the temporal information across frames, leading to temporally inconsistent scene description.

Detecting occlusion boundaries is a well studied problem, due to its usefulness in understanding the depth, motion and context of the scene [22, 12]. Fleet *et al.* [6] gave a Bayesian formulation where boundaries resulted from distinguishing local image motion. Stein *et al.* [22] has shown that combining appearance and motion cues improves occlusion boundary detection. They further improve occlusion boundary detection by applying a global conditional random field where the potentials are learned from AdaBoost. He *et al.*[9] showed that a global model may not be necessary for sequences with ego-motion and achieved comparable results by local edge and psuedo-depth maps. Recently, Sundberg *et al.* [23] improved over these boundaries by computing motion gradients across static boundaries. Since these methods rely on local features they are unable to reduce false positives where intra-object local motion or appearance variance is high. Typical examples include waves in the water or trees in the wind. In our method, semantic/geometric labels help suppress such errors.

Other methods have also been proposed to detect occlusion boundaries in a single image. Many methods inferring geometric labels initially estimate boundaries in single images [20, 19, 7]. Probabilistic boundary detectors like Pb [16] use local oriented energy, color, and texture gradients. Arbeláez *et al.* [1] improve boundary detection by imposing global constraints via spectral clustering which results in closed contours. Leordeanu *et al.* [14] proposed *Gb*, which reduces the time for generalized boundary detection by efficient computing boundary normals. In the last year, probabilistic boundaries have become feasible to use for realtime applications. The first method that deserves mention is Sketch Tokens [15], which classifies edge patches using a random forest. Following this work, Dollár and Zitnick [5] introduced a realtime structure learning method for edge detection. From our point of view, both of these methods make many leaps forward in the single image boundary detection problem. Yet, extending these methods to videos is a nontrivial challenge. In this paper we compare to both Sketch Tokens, and the single-scale (SE-SS_T4). and multi-scale (SE-MS_T4) version of Structured Edges. Unlike previous methods, we use geometric or semantic labels over video segmentation for finding temporal consistent boundaries in videos.

In this paper, we leverage video segmentation to find temporally consistent boundaries in dynamic scenes. We use flow, and geometric features for estimating each edgelet's occlusion probability, and then enforce boundary continuity in a pairwise MRF framework. We demonstrate the importance of temporal smoothing and geometric features in occlusion boundary estimation. To verify our claims, we developed a comprehensive video occlusion boundary ground truth dataset with a broad set of examples.

3. Dataset and Annotation

Existing Datasets: A comprehensive dataset is necessary for evaluation of temporal occlusion boundary detection in dynamic scenes. However, existing datasets are limited to ground truth annotation for intermittent frames in a sequence [3], and not all include semantic information. This poses a hurdle in the study of the role of the scene structure, and temporal dynamics for occlusion reasoning. Two widely used datasets for occlusion detection in videos are proposed by Stein *et al.* [22] and Sundberg *et al.* [23]. These datasets are limited in their scope as (1) they provide ground-truth for only a single frame; (2) they are not suitable for the study of the role of the scene layout in occlusion reasoning and were not developed for that purpose. Butler *et al.* [4] developed MPI-Sintel flow dataset which contains motion boundaries but does not include the occlusion boundaries in static background. The only dataset with semantic labels, and occlusion boundaries was proposed by Hoiem *et al.* [11]. They proposed a dataset of 50 images

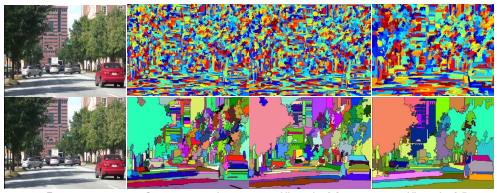

Figure 2. Video segmentation from Xu *et al.* [26] at the top row and from Grundmann *et al.* [8] at the bottom row. We selected the over-segmentation (hierarchy level=0) from Grundmann *et al.* [8] because of its performance in preserving occlusion boundaries and longer temporal coherence over our challenging dataset.

with ground truth annotation of outdoor scenes with occlusion boundaries, surface layout, and depth order. Since this dataset contains only a single image for a scene, it is also not ideal for our study. To overcome this limitation, we have developed a comprehensive dataset with temporal occlusion boundaries and semantic annotations.

A Video Dataset for Occlusion Reasoning: Our dataset consists of 30 outdoor videos of urban scenes. Some videos were recorded while walking, some while driving, and others were downloaded from YouTube. We also included few videos from recently released video geometric context dataset from Raza *et al.*[17]. The videos contain sky, ground, roads, pavements, rivers, buildings, trees, humans, and cars. Annotating temporal occlusion boundaries is a challenging task and there has been no such dataset until now. We annotate temporal occlusion boundaries in videos by using video segmentation, similar to the approach by Hoiem *et al.*to annotate image dataset using super-pixels [11]. Recently, two video segmentation algorithms have been proposed [8, 26]. Both these algorithms provide a hierarchy of segments from a video. We show a video and segmentation hierarchy output of these algorithms in Figure 2. The algorithm by Xu *et al.*gives a high number of super-voxels with very short temporal life, while the output from Grundmann *et al.*gives less super-voxels with longer temporal life as well as preserving the occlusion boundaries. We therefore selected the video segmentation algorithm proposed by Grundmann *et al.*[8] to annotate temporal occlusion boundaries.

We use [17, 8] to annotate the video [1]. We group together the spatio-temporal super-voxels that belong to individual objects, as well as geometric class labels. Geometric classes annotated in our dataset are sky, ground (roads, pavements, grass, rivers), planer surfaces (buildings and rocks), porous (trees and foliage), and movable objects (humans, cars, and trains). Boundaries between individual objects and geometric classes provide occlusion boundaries. We show the process of our manual occlusion boundary annotation in Figure 3. In very few cases, the segmentation algorithm fails to segment a region due to similarity in color or poor-lighting condition. These boundaries are not annotated in our dataset but such cases are only a small fraction of the whole dataset. The proposed dataset contains 5,042 annotated frames across 30 videos. Table 1 provides a comparison of our dataset with existing datasets.

[1] www.videosegmentation.com

Figure 3. Occlusion boundary annotation: (a) an input video, (b) spatio-temporal super-voxels, (c) we cluster super-voxels into semantic classes and objects, (d) boundaries between these semantic classes and objects give occlusion boundaries.

Name	Image/Video	Ground-truth frames	Semantic Labels
CMU Geometric Context [11]	Image	50	Yes
CMU Occlusion [21]	Video	30	No
BSDS [23]	Video	60	No
Ours	Video	5042	Yes

Table 1. Comparison with existing datasets providing ground truth for occlusion boundaries. Our dataset contains 5042 frame across 30 videos with annotations for occlusion boundaries, and geometric class labels.

4. Approach

We provide an overview of our proposed method as shown in Figure 4. We begin by over-segmenting the video into spatio-temporal super-voxels. Then, for each neighboring region pair, we extract features to characterize the edgelet between those regions. In particular, we leverage geometric context features to consider the semantic layout in occlusion boundary detection. First, we train a binary classifier to estimate the probability for an edgelet to lie on an occlusion boundary. Next, we enforce occlusion boundary continuity in MRF model by using pairwise edgelet occlusion boundary probability learned by a separate classifier. Finally, we perform temporal smoothing of these estimated occlusion probabilities by aggregating them across successive frames. We perform detailed experiments to show the importance of geometric context features and temporal smoothing for predicting occlusion boundaries in videos. In the following, we describe each step of our algorithm in detail.

4.1. Video Segmentation

We build our algorithm on the initial boundaries provided by video over-segmentation. The purpose of using video segmentation is to find spatio-temporal regions which are coherent in appearance and motion. We use the video segmentation algorithm (and the related online system) proposed by Grundmann *et al.*[8] and its extensions [25]. There method's over-segmentation gives a large number of spatio-temporal super-voxels, which we use as initial candidates for occlusion boundaries. Classifying occlusions on over-segmentation boundaries has following advantages: (1) pro-

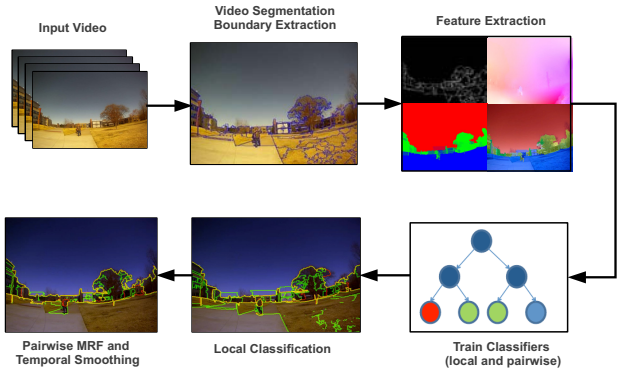

Figure 4. Overview of our method. We learn occlusion boundaries in a pairwise edgelet MRF framework using unary and continuity occlusion boundary probabilities using edgelet, flow, and geometric features. Then we temporally aggregate the frame by frame predictions to remove inconsistent boundaries.

vides good candidate locations for occlusion detection; (2) it reduces the complexity of the algorithm by not having to classify individual pixels; (3) by working with supervoxels we can enforces temporal coherence or occlusion boundaries; (4) helps in exploiting temporal redundancy and causality for efficient video processing. Next we develop models for learning occlusion boundaries using these candidate boundaries.

4.2. Features for Occlusion Boundary Prediction

To train classifiers for occlusion boundary prediction, we compute a variety of features. Features are computed on every frame for each edgelet, *i.e.*, boundary between two regions. An edgelet might span more than one frame, in which case it will contribute to the training data multiple times. For each edgelet, we compute features based on boundary, regions, flow, and geometric context. These features are explained next.

Boundary and Region Based Features Segmentation boundaries provide good candidate locations for finding occlusion boundaries. Longer boundaries with strong color gradients are more likely to be occlusion boundaries as compared to weak short boundaries, we compute boundary length and smoothness for each edgelet, as suggested by [11]. In addition to the boundary features, we also include the color difference of the regions surrounding the edgelet.

Optical-flow/Motion Based Features Motion estimates may have inconsistencies at the occlusion boundaries due to parallax. To capture this information in our framework we compute optical flow based features at each edgelet. We compute optical flow using the total variation method proposed by Wedel *et al.*[24]. Flow algorithms have photo-consistency assumption. Therefore, pixels advected from reference frame I_t by estimated flow $F_{t\to t+1}$ should correspond to the next frame I_{t+1}. This assumption breaks down at occlusion boundaries, hence high photo-consistency residual should be indicative of such boundaries [12, 13]. Residual photo-consistency feature $\mathcal{F}_{\text{PC}}$ is computed as

$$\mathcal{F}_{\text{PC}}(x) \;=\; \left|I_t(x) - I_{t+1}(x + F_{t\to t+1}(x))\right|. \qquad (1)$$

If the motion of two interacting objects is different, their occlusion boundary will have flow discontinuities. To include flow discontinuities, we compute the flow gradient given by

$$\mathcal{F}_{\text{TG},x} \;=\; \| \bigtriangledown u_x \|, \quad \mathcal{F}_{\text{TG},y} \;=\; \| \bigtriangledown v_y \|. \qquad (2)$$

Since flow gradient is only computed over two pixels, it is unable to capture statistics over a larger area. To capture these proximal flow discontinuities, we compute the variance of the magnitude of flow $F_{\overrightarrow{\text{mag}}} = \|F_{t\to t+1}\|$ in a spatial window around a pixel given as

$$\mathcal{F}_{\text{mag}}(x) \;=\; \mathbf{E}\left[\left(F_{\overrightarrow{\text{mag}}}(x_i) - \mathbf{E}\left[F_{\overrightarrow{\text{mag}}}(x)\right]\right)^2\right], \qquad (3)$$

where x_i are the pixels in the 3×3 window around pixel x and $\mathbf{E}(\cdot)$ is the expectation function. Another way to check inconsistency in flow is to advect pixels by flow $F_{t\to t+1}$ and follow them back by flow $F_{t+1\to t}$, *i.e.*, flow computed from I_{t+1} to I_t. If the pixel was not occluded or dis-occluded, *i.e.*, it was far from an occlusion boundary, an accurate flow estimate should bring the pixel back to its starting location in frame I_t. We use the ℓ_2 distance from the starting location as a reverse flow constancy feature,

$$\mathcal{F}_{\text{RC}} \;=\; \|x - (x'_F + F_{t+1\to t}(x'_F))\,\|, \qquad (4)$$

where $x'_F \;=\; \text{round}(x + F_{t+1\to t}(x))$. We can similarly note the inconsistency in the forward and reverse flow angle. $F_{t\to t+1}$ and $F_{t+1\to t}$ are said to be consistent if they are 180° apart. Any deviation from this is used as a reverse flow angle consistency feature, which is computed as,

$$\mathcal{F}_{\text{RC},\theta} \;=\; \left|\pi - \arccos\left[\frac{F_{t\to t+1}(x)\cdot F_{t+1\to t}(x'_F)}{F_{\overrightarrow{\text{mag}}}(x)F_{\overleftarrow{\text{mag}}}(x'_F)}\right]\right|, \qquad (5)$$

where $F_{\overleftarrow{\text{mag}}} \;=\; \|F_{t+1\to t}\|$ is the magnitude of the reverse optical flow.

Geometric Layout Features Geometric layout estimate provides strong cues for occlusion boundaries and have been shown to be useful for occlusion reasoning and scene understanding [10]. For example, an occlusion boundary should exist between different geometric classes, such as, between sky and vertical class(buildings, trees, etc). To include geometric layout estimate for dynamic video scenes, we use the method proposed by Raza *et al.*[17]. Their method provides confidence for each pixel belonging to geometric classes, such as sky, ground, static-solid, porous,

and movable-objects. We use the most likely geometric label and the difference of the average confidence of each geometric class of neighboring regions as feature for occlusion reasoning.

4.3. MRF Model

Our goal is to maximize the probability of an edgelet e being an occlusion boundary given the edgelet feature vector, *i.e.*, $P(e = \text{Occlusion}|X)$. We can estimate this probability in MRF model as,

$$P(e = \text{Occlusion}|X) = \frac{1}{Z}\prod_{n=1}^{N} g_n(e_n, X_n) \prod_{m \in Conn.(n)} f_{mn}(e_n, e_m) \quad (6)$$

where $g_n(\cdot)$ is the unary probability of an edgelet being an occlusion boundary, and $f_{mn}(\cdot,\cdot)$ is the pairwise term capturing the occlusion probability for an edgelet with relation to its connected edgelets. The unary term $g_n(\cdot)$, the occlusion boundary probability of an edgelet, is computed by training a random forest classifier. Random forest are useful for their performance on learning high dimensional non-linear relationships, while providing feature selection and importance for free [2, 18]. We trained random forest with 105 trees, 11 random features per node, and a maximum depth of 35 nodes for a tree. We train the unary classifier with the features from each edgelet of each frame to capture the temporal variations.

The unary classifier, computes the probability of an individual edgelet to be an occlusion boundary edgelet. To enforce continuity of occlusion boundaries, we train a separate random forest classifier to estimate the pairwise edgelet probability. For continuity classifier, we compute the feature for each edgelet pair by concatenating the unary features of both the individual edgelets. The positive pairwise occlusion boundaries are the examples with both edgelets having ground truth occlusion boundary label "true".

To predict occlusion boundaries for a test video, we compute occlusion features for each edge of each frame in the over-segmented video. Then we compute the unary and pairwise occlusion boundary probabilities. Final occlusion probability for an edgelet is computed using the MRF model given in Equation (6). We use loopy-belief propagation algorithm to find the approximate solution for Equation (6). Pair-wise continuity MRF model reduces false positives over unary occlusion boundary estimate, as shown in Figure 5. Now, we have assigned each edgelet an occlusion probability and thresholding these probabilities would give occlusion boundaries. However, these estimates may be temporally inconsistent, *i.e.*, occlusion probability of an edgelet may change significantly from one frame to the next.

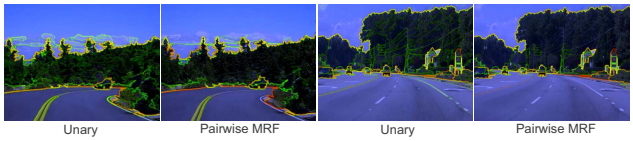

Figure 5. Pairwise MRF reduces the false positives in unary prediction as shown above. The yellow, green, and red boundaries show true positives, false positives, and false negatives, respectively.

To provide temporal consistent occlusion boundaries, we again leverage from video segmentation to temporally smooth the occlusion probability of an edge over a temporal window. The temporal window starts where an edgelet is first formed by two neighboring spatio-temporal regions. Once we have processed the number of instances of a unique spatio-temporal edgelet equal to the length of temporal window, we average the occlusion boundary probabilities in the temporal window for that edgelet, and ignore all future instances of that edgelet. This results in an occlusion boundary algorithm which is linear to the number of unique edgelets in a video than the algorithms which treat video as individual frames and have a complexity of number of edgelets $\times$ number of frames. We experiment with different lengths of temporal windows to filter out temporally inconsistent boundaries (Section 5).

5. Results

In this section, we report the quantitative and qualitative results of our algorithm. Specifically, we measure the performance of our method as precision *vs.* recall (PR) curves estimated over 5-fold cross-validation by varying the threshold. To compute the precision *vs.* recall curve for our experiments with temporal smoothing, we choose the temporal window with maximum F-1 measure. In our experiments, the occlusion boundary prediction performance becomes stable for a temporal window of size greater than 15 frames. The plot in Figure 6 shows that geometric features combined with temporal smoothing results in the best performance. Also, note that temporal smoothing improves performance for each feature set. Table 2 shows F-1 measure of each case.

Our results show qualitative improvement in occlusion boundary detection using geometric context (please see supplementary video). In Figure 7, we show the importance of each feature set from the random forest's out-of-bag training estimate. It is evident from the bar-plot, that geometric features provide more discriminative information for occlusion boundary detection. We show examples to verify the importance of these geometric features, in Figure 8. Note, that the inclusion of geometric features improves occlusion boundary detection by removing boundaries within a geometric class, *e.g.*, boundaries appearing on the ground, across sky, or within trees. Further, they provide important cues to enforce a boundary between different geometric classes.

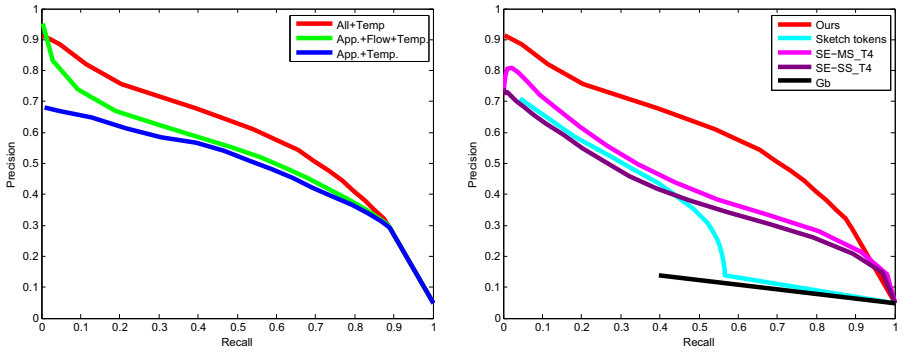

Figure 6. Performance evaluation: Precision *vs.* recall (PR) curves for occlusion boundary detection on our dataset. For our algorithm, we used a temporal window of 30 frames. Legend: ALL (appearance+flow+geometric features), App (appearance features only), and Temp (with temporal smoothing). (Left) Results show that geometric features combined with temporal smoothing yields in the best performance compared to other feature combinations.(Right) Comparison of our method with Sketch Tokens [15], SE-MS_T4 [5], SE-SS_T4 [5], and Gb [14].

Features	**Ind. Frames**	**Temporal**
ALL	0.58	0.60
Appearance+Flow	0.53	0.55
Appearance Only	0.52	0.53

Table 2. Comparison of feature sets by F-1 measure. Appearance only uses "Boundary and Region Based Cues". Appearance+Flow adds to it "Optical-Flow Based Cues." Individual frame-based processing considers each frame individually in the video, whereas the temporal approach takes advantage of causality in videos, by processing over a 30 frame temporal window.

Algorithm	**F-1**
Ours	**0.60**
Sketch Tokens [15]	0.42
SE-MS_T4 [5]	0.46
SE-SS_T4 [5]	0.43
Gb [14]	0.21

Table 3. Performance comparison of our method with existing algorithms. Our method exploits causality in videos for temporal occlusion boundary detection.

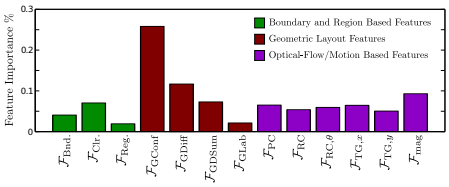

Figure 7. Feature importance estimate from random forest over 5-fold cross-validation. The bar-plot shows the number of votes casted by each feature for the correct class in out-of-bag estimate [2]. Geometric confidence $\mathcal{F}_{\text{GConf}}$ estimate of the neighboring regions stands out as most useful along with their difference $\mathcal{F}_{\text{GDiff}}$, and the absolute sum $\mathcal{F}_{\text{GDSum}}$. Other useful features are flow magnitude variance $\mathcal{F}_{\text{mag}}$, photo-consistency $\mathcal{F}_{\text{PC}}$, and color feature $\mathcal{F}_{\text{Clr}}$.

Some misclassification results are shown in Figure 9. A reason for occlusion boundary misclassification is that we have a very challenging dataset with fast jittery motion. Spatio-temporal segments tend to break quickly in such videos, resulting in very short lived temporal boundaries. For these boundaries temporal smoothing is not effective. Some mis-classifications also occur in shadows due to bad lighting conditions.

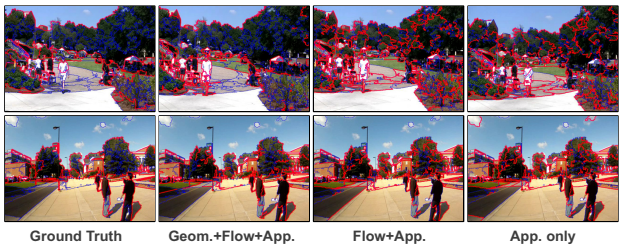

Figure 8. Qualitative analysis of feature importance: Figure shows qualitatively that geometric features improve accuracy significantly. Visual comparison is performed with a temporal window of size = 30, and threshold is selected at the peak of F-1 measure. Occlusion and non-occlusion boundaries are shown in red and blue, respectively.

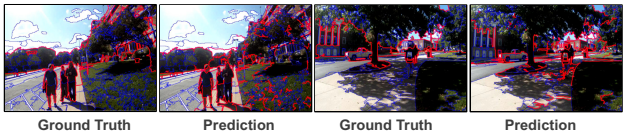

Figure 9. Failure cases. Occlusion boundaries are mis-predicted due to shades and fast jittering movement. Temporal smoothing is not useful in fast jittery motion sequences due to short temporal life of segments.

Direct comparisons and evaluations to other efforts and datasets, with quantitative measures, is hard for our work as there is no such dataset with temporal occlusion boundary, and semantic label annotations (see Section 3). In any case, we do undertake and provide a comparison with the occlusion boundaries detected with other occlusion boundary detection algorithms[14, 15, 5]. We applied their publicly available code on our dataset. Table 3 and Figure 6 (Right) shows the comparison of our method with the ex-

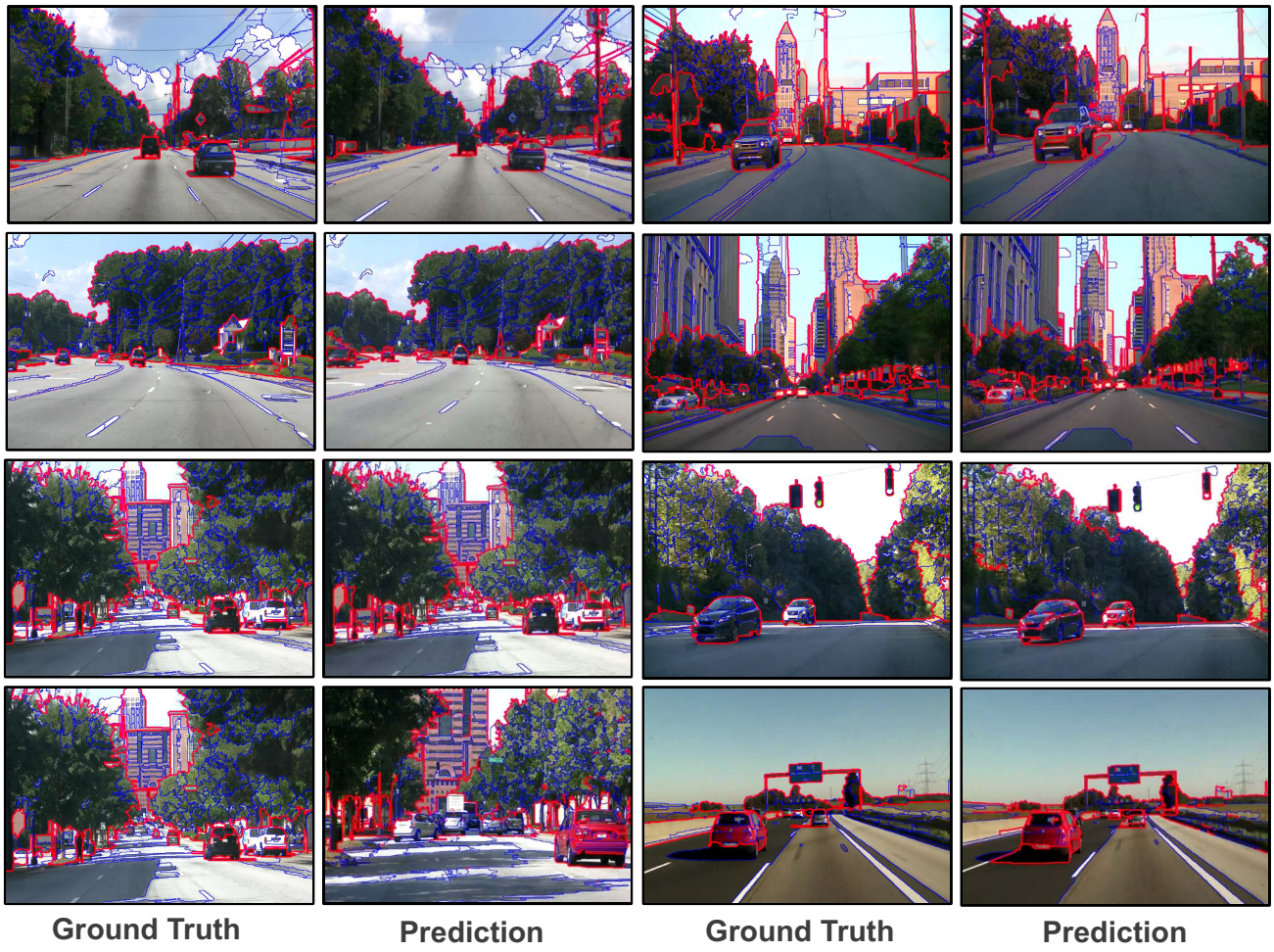

Figure 11. Qualitative results for occlusion boundary prediction: (left) Ground truth, (right) Predicted occlusion boundaries using geometric, flow, and appearance features with temporal smoothing (temporal window size=30). Occlusion and non-occlusion boundaries are shown in red and blue, respectively.

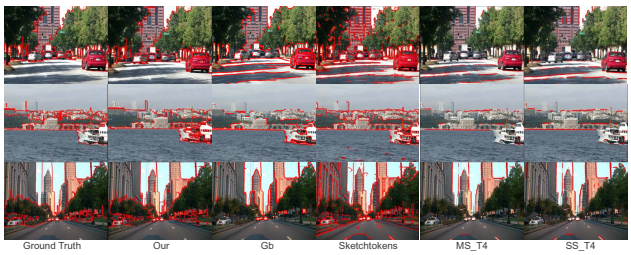

Figure 10. Qualitative comparison of occlusion boundaries predicted by *Gb* [14], Sketch Tokens [15], and multi-scale (SE-MS_T4) and single scale (SE-SS_T4) Structured Edges [5]. The probabilistic boundaries are thresholded using the best F-1 score over all sequences.

isting algorithms. To compensate for occlusion boundaries detected by different algorithms in proximity of our ground-truth, we dilate our boundary labelling by a pixel (*i.e.*, an error margin of 3 pixels). We achieve better performance as compared to other methods. Our algorithm can avoid making false-positive detection within a geometric class, *e.g.*, within tree regions, or boundaries on the ground but other algorithms lack this ability. In addition, by leveraging spatio-temporal occlusion boundaries, we can learn features from all the temporal samples of occlusion boundaries. Figure 10 shows qualitative comparison of the above comparison by overlaying occlusion boundaries threholded at best F-1 score. It shows that our algorithm can detect occlusion boundaries between different geometric classes, and avoid false positives within a geometric class. *Sketch tokens* algorithm detects most of the boundaries as occlusion boundaries, while *Gb*, *SE-MS_T4*, and *SE-SS_T4* detect less boundaries with very few false positives. It should be noted that in our temporal occlusion boundary detection approach, we exploit causality to process the videos efficiently. Temporal occlusion boundary detection only requires T (*i.e.*, length of temporal window) samples of each unique boundary but other approaches require processing the whole video

sequence. Operating on a temporal window makes it possible for our algorithm to be be applied to streaming video approaches. Figure 11 shows more qualitative results of our approach.

6. Conclusion

We have presented an approach for finding temporally consistent occlusion boundaries in dynamic outdoor scenes. We learn occlusion boundaries using edge, flow, and geometric context based features in a pairwise edgelet continuity MRF model. The results are computed on the spatio-temporal boundaries provided by over-segmentation [8]. We choose graph-base video segmentation algorithm for its accuracy in preserving occlusion boudaries, temporal coherence, and ability to handle long video sequences efficiently. However, our approach for learning occlusion boundaries is independent of any particular video segmentation algorithm and should perform well using other video over-segmentation algorithms. The results in this study demonstrate the importance and benefit of integrating scene layout for occlusion reasoning. Moreover, we show that temporal smoothing improves accuracy over independent frame-by-frame processing. Our proposed algorithm also processes videos efficiently by exploiting causality and temporal redundancy using spatio-temporal video segmentation. We have also developed a comprehensive dataset with ground truth temporal occlusion boundary annotations and a broad set of examples containing dynamic scenes. In the future, we plan to integrate more semantic classes and depth information in our method.

Acknowledgement This material is based in part on research by the Defense Advanced Research Projects Agency (DARPA) under Contract No. W31P4Q-10-C-0214, and by a Google Grant and and Google PhD Fellowship for Matthias Grundman, who participated in this research as a Graduate Student at Georgia Tech. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of any of the sponsors funding this research.

References

[1] P. Arbeláez, M. Maire, C. Fowlkes, and J. Malik. Contour detection and hierarchical image segmentation. *PAMI*, 33(5):898–916, 2011.

[2] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[3] G. J. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla. Segmentation and recognition using structure from motion point clouds. In *ECCV (1)*, pages 44–57, 2008.

[4] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In *European Conf. on Computer Vision (ECCV)*, 2012.

[5] P. Dollár and C. L. Zitnick. Structured forests for fast edge detection. In *IEEE ICCV*, 2013.

[6] D. Fleet, M. Black, and O. Nestares. Bayesian inference of visual motion boundaries. *Exploring artificial intelligence in the new millennium*, pages 139–173, 2003.

[7] S. Gould, R. Fulton, and D. Koller. Decomposing a scene into geometric and semantically consistent regions. In *ICCV*, 2009.

[8] M. Grundmann, V. Kwatra, M. Han, and I. Essa. Efficient hierarchical graph-based video segmentation. In *IEEE CVPR*, 2010.

[9] X. He and A. Yuille. Occlusion boundary detection using pseudo-depth. In *ECCV*, pages 539–552. Springer, 2010.

[10] D. Hoiem, A. Efros, and M. Hebert. Closing the loop in scene interpretation. In *IEEE CVPR*.

[11] D. Hoiem, A. Stein, A. Efros, and M. Hebert. Recovering occlusion boundaries from a single image. In *ICCV*, pages 1–8, 2007.

[12] A. Humayun, O. Mac Aodha, and G. Brostow. Learning to find occlusion regions. In *IEEE CVPR*, 2011.

[13] S. Ince and J. Konrad. Occlusion-aware optical flow estimation. *Image Processing, IEEE Transactions on*, 17(8):1443–1451, 2008.

[14] M. Leordeanu, R. Sukthankar, and C. Sminchisescu. Efficient closed-form solution to generalized boundary detection. In *ECCV 2012*, pages 516–529. Springer, 2012.

[15] J. J. Lim, C. L. Zitnick, and P. Dollar. Sketch tokens: A learned mid-level representation for contour and object detection. In *IEEE CVPR*, 2013.

[16] D. R. Martin, C. C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE PAMI*, 26(5):530 –549, may 2004.

[17] S. Raza, M. Grundmann, and I. Essa. Geometric context from video. In *IEEE CVPR*, 2013.

[18] Y. Saeys, T. Abeel, and Y. Van de Peer. Robust feature selection using ensemble feature selection techniques. In *Machine Learning and Knowledge Discovery in Databases*, pages 313–325. Springer, 2008.

[19] A. Saxena, S. Chung, and A. Ng. 3-D Depth Reconstruction from a Single Still image. *IJCV*, 76(1):53–69, 2008.

[20] A. Saxena, M. Sun, and A. Y. Ng. Make3d: Learning 3d scene structure from a single still image. *PAMI*, 31(5):824–840, 2009.

[21] A. Stein and M. Hebert. Combining local appearance and motion cues for occlusion boundary detection. *Robotics Institute*, page 349, 2007.

[22] A. Stein and M. Hebert. Occlusion boundaries from motion: low-level detection and mid-level reasoning. *IJCV*, 2007.

[23] P. Sundberg, T. Brox, M. Maire, P. Arbeláez, and J. Malik. Occlusion boundary detection and figure/ground assignment from optical flow. In *IEEE CVPR*, 2011.

[24] A. Wedel, T. Pock, C. Zach, H. Bischof, and D. Cremers. An improved algorithm for tv-l 1 optical flow. In *Statistical and Geometrical Approaches to Visual Motion Analysis*, pages 23–45. Springer, 2009.

[25] C. Xu and J. Corso. Evaluation of super-voxel methods for early video processing. In *IEEE CVPR*, 2012.

[26] C. Xu, C. Xiong, and J. J. Corso. Streaming hierarchical video segmentation. In *ECCV*, 2012.