1 Introduction

In this chapter, we present a method, illustrated in Fig. **??**, to endow a scene, densely reconstructed from monocular video, with a metric that incorporates geometric homogeneity and image topology through occlusions. While the latter are temporally inconsistent (they change with the video), the way they change is spatially consistent, an observation key to defining affinities that allow us to partition the scene into coherent "objects" at a level of granularity relevant to the viewer. Occlusions inform the scale of the segmentation, allowing the selection of a partitioning of the scene, out of all possible partitions, that respects the occlusions present in the video. For robotic interaction tasks, such as manipulation or obstacle avoidance, the granularity of the scene representation can be critical, and our approach focuses the scene segmentation task on objects that generate occlusions in the images due the motion of the viewer relative to the scene.

To achieve this, we employ a robust metric on the scene using a combination of curvature-based geodesics on a 3D mesh and back-projected occlusion-constrained image segmentations. The spatial consistency of these segmentations on the scene allows our segmentation method to adapt to the scale informed by the occlusions in the video. While one could employ trained object detectors at the outset to arrive at a semantic segmentation of the scene, we focus on low-level geometric and topological cues first, to segment the scene and images into coherent regions, where one could then deploy object detectors if so desired. Semantic analysis of the scene involves object identities and relations, and knowledge of scene geometry, topology, and putative object regions are key to infer the latter. This is the focus of our work.

The remainder of the chapter is organized as follows: In Sec. 2.1 we formulate scene (and video) segmentation as a selection problem on the set of potential nested partitions of the underlying scene based on homogeneity properties, and Sec. 2.3.1 presents occlusion-based image segmentation as a scale-selection mechanism on this set of potential partitions. Sec. 2.2.2 presents the construction of a curvature-augmented geometry on a 3D mesh used in Sec. 2.3.2 to regularize these occlusion-based selections obtained from the image frames through an adaptive geodesic that combines these two components to perform a scene segmentation at the level of granularity relevant to the viewer. Finally, Sec. 3 shows a quantitative and qualitative evaluation of our scale-adaptive segmentation scheme on a ground-truthed dataset of monocular dense reconstructions that we have collected.

The work described here is the result of a fifty-fifty collaboration with Konstantine Tsotsos, who designed and assembled the 3D pipeline, made significant contributions to the "object distance" metric, and implemented the segmentation scheme.

1.1 Contributions and Related Work

There is a vast body of literature pertaining to semantic segmentation of *images* **??????????**; our work is particularly related to *joint* segmentation (a.k.a. "co-segmentation") of multiple images, or *video* **??????**. However, the goal of such approaches is a partitioning of the spatio-temporal image volume, not of the *scene* that generated it. Seeking a segmentation of the scene allows

us to bypass the complex and discontinuous changes in the partitioning of the video due to scale, spatial quantization, and occlusions. Furthermore, occlusions provide local ordering constraints that can be used to partition the image into "layers" **??????** by solving a convex optimization problem **?**. In all of these cases, "objects" are collections of pixels that are often *temporally inconsistent* as the local ordering constraints can change over time (think of a merry go round), producing flickering segmentations. However, by by using occlusions as topological cues these segmentations tend to correspond to spatially consistent regions on the scene, even if their image labels are not temporally consistent. Our method relies heavily on this observation to accumulate data-driven cues for the extent of individual objects in the scene.

Our work can be interpreted as an attempt to combine multiple image segmentations (which depend on both the scene and the viewer) into one persistent segmentation of the scene independent of the viewer. Since our method involves the intermediate reconstruction of a dense three-dimensional model of the scene, which we do in real time, our work also relates to multiple-view stereo and structure-from-motion, and in particular real-time dense multi-view stereo **???**. As an alternative, one could use an alternate range sensor **?**, for instance based on structured light **?**, although those perform poorly in natural illumination and have a fixed scale of interaction.

There are relatively few attempts to generate a dense label field in the *scene* **?**. While semantic labels have been attached to various forms of 3D reconstruction, these typically are *sparse* (*e.g.,* collections of feature descriptors and their coarse positions **??**).

Our work is also related to **?**, in which Regression and Decision Tree Fields are used to segment a 3D scene, and **?**, in which SVMs are used to segment point clouds gathered from RGB-D data. Similarly, Bleyer et al. **?** describe a method for labeling that is explicitly compatible with the 3D structure of the scene. Although direct comparison with these algorithms is not possible as neither their code nor their datasets are publicly available, in Sect. 3.2 we report experiments on data similar in nature and scale that we intend to release publicly upon completion of the anonymous review process. Other related work includes **???**, where the focus is on manipulation. Additionally, Zheng et al. **?** use 3D point clouds to find objects in the scene using geometric and physical cues from RGB-D data. Most closely related to our work are are the recent works of **????**), all of which generate semantic segmentations of the scene using responses from trained detectors as input. We seek to generate segmentations at a similar level of abstraction (albeit without semantic labels) through viewpoint-based topological homogeneity instead of semantic homogeneity (object detector responses). Our method can also be thought of as a generic proposal scheme for regions within which to collect support for semantic categorization based on geometric and viewpoint based contextual information. Partitions based on solely geometric homogeneity have also been explored extensively by the 3D mesh segmentation community (**???**).

Our contributions are (a) a method for scene segmentation leveraging the spatial consistency of temporally inconsistent image cues, (b) an adaptive geodesic distance function on the scene shaped by spatially consistent image cues in the form of occlusions, (c) an object-level scene segmentation scheme that extends a real-time dense reconstruction system based on monocular video. To compare with standard approaches for segmenting dense geometry, we have (d) collected

a calibrated dataset with a variety of objects of different scales and textures, indoor and outdoor, on natural and artificial laboratory scenes. A key assumption for our dense monocular reconstruction pipeline is that the only thing moving in the scene is the viewer. Extension beyond cases where this assumption holds is desirable, but even the static case is relevant to several applications from robotic inspection to autonomous navigation and exploration.

2 Methodology

2.1 Scene Model

The input to our system is a grayscale video $\{I_t\}_{t=0}^{T}$, with each image I_t mapping from a domain $D \subset \mathbb{R}^2$ to $\mathbb{R}_+$. The desired output is a constant partitioning of a higher-dimensional "scene" that the video observes, from which we can also derive a piecewise-constant, integer-valued function $c_t(x)$ that associates to each pixel $x \in D$ a label. The scene is represented by a (multiply-connected) collection of surfaces $S \subset \mathbb{R}^3$ supporting a reflectance function (albedo) $\rho : S \to \mathbb{R}_+$. Under the Lambert-Ambient model, the image and the scene are related by

$$\begin{cases} I_t(x) = \rho(p) + n_t(x), \ p \in S \\ x = \pi(g_t p) + v_t(x) \end{cases} \tag{1}$$

where $g_t \equiv (R(t), T(t)) \in SE(3)$ is the pose of the camera relative to the reference frame of S, and $\pi : \mathbb{R}^3 \to D$ is a canonical central (perspective) projection. The residual $n_t(x)$ accounts for unmodeled photometric phenomena such as changes in illumination (assumed negligible in the short time-span during which the video is captured), deviations from Lambertian reflection, sensor noise etc. The residual $v_t(x)$ accounts for violations of the geometric assumptions (rigid motion, static scene). Estimates $\hat{g}_t, \hat{S}$ are obtained as described in Sec. 2.2.1.

We consider that objects in the scene compose a nested covering of sets $\mathfrak{S} = \{S_i\}_{i=1}^{K}$, where $S_i \cap S_j \in \{\emptyset, S_i, S_j\}$ for all i and j, and $\bigcup_{i=1}^{K} S_i = S$. Any segmentation of the scene is a selection $\mathcal{P} = \{S_{\mathcal{P},i}\}_{i=1}^{K_\mathcal{P}}$ of disjoint sets in $\mathfrak{S}$ such that $\bigcup_{i=1}^{K_p} S_{\mathcal{P},i} = S$. For a partitioning $\mathcal{P}$ to be meaningful, typically some homogeneity property must hold on each $S_{\mathcal{P},i}$, be that geometric, photometric, semantic, topological, or some combination.

One can perform a sequence of still-frame (or short-baseline video) segmentations $c_t : D \to \mathbb{Z}^+$ using a subset of these properties which can then be leveraged into a segmentation of the scene. A reasonable such segmentation is one that does not oversegment the scene relative to c_t (distinguish points that have the same label c_t for all or almost all t), or undersegment (fail to distinguish points that tend to have different labels), more than necessary. As these c_t will be temporally inconsistent, they can be regularized by integration on the scene using geometric homogeneity. We consider a particular set of segmentations c_t to induce a selection $\mathcal{P}$ from $\mathfrak{S}$. The use of occlusion-based image segmentation (Sec. 2.3.1) to induce a segmentation respecting topological homogeneity as seen by the viewer is a key contribution of our approach.

2.2 Curvature Augmented Geometry and Geometric Affinity

Here we discuss the construction of a 3D mesh representation of the scene, and of an augmented geometry for curvature-based affinity computations.

2.2.1 Dense Monocular Reconstruction

A estimate of the scene $\hat{S}$ is reconstructed in an on-line fashion as the camera browses the scene. We use the real-time camera tracking system PTAM **?**, a fast dense stereo module, and a globally optimal depth map fusion algorithm. The latter component takes as input depth maps and camera poses obtained from the former components, and computes a dense surface using an implicit volumetric representation via a truncated signed distance function (TSDF), similar to **???**. Dense depth maps are computed using multiview plane-sweeping **?** on a set of images and their camera poses obtained from the tracking module. The main advantage of this approach is that it allows for arbitrary scene topology. It is also closely related to the Kinect Fusion **?**, although we do not employ a depth sensor but work solely based on image data. Example surface normal and depth maps extracted from our dense reconstruction are shown in figure 1.

For the purposes of computing affinities between points on $\hat{S}$, we construct discretized mesh derived from the regular voxelization of the reconstructed scene. Affinities between regions of the scene can be computed as functions of the nodes of this mesh. Each node q aggregates the spatial information (mean location, mean normal, Sec. 2.2.2) and image-based topological cues (Sec. 2.3.1) of the surfaces passing through the associated voxel.

2.2.2 Computing Geometric Affinity

Affinities (or distances) between regions of the scene can be computed as functions of the nodes of this mesh. Following standard geometric mesh segmentation schemes (**???**) we use surface curvature as a heuristic for partitioning contiguous mesh regions. In particular, we try to cut the mesh at "creases", regions where one of the principal curvature directions dominates the other. At each mesh node, we compute the local principal curvatures $k_1(q) > k_2(q)$, and the corresponding principal directions $v_1(q)$ and $v_2(q)$, by fitting a second-order surface to that node and its neighbors. The scalar field $K(q) := \max\{0, \frac{k_1(q)^2}{k_1(q)+|k_2(q)|}\}$ computed at every mesh node measures the strength and dominance of the most-positive eigenvalue, $k_1(q)$ (see Figure 2). The augmented geometric distance between two points q_i and q_j on the mesh is computed as a K-weighted mesh geodesic, that is, as the minimum path length $d_G(q_i, q_j) = \min_{\{\substack{s_0 \to \cdots \to s_n \\ q_i = s_0,\ q_j = s_n}\}} d_G(s_0, \ldots, s_n)$, where the $s_0, \ldots, s_n$ are a sequence of connected intervening nodes, $\{A_1, A_2\}$ are scalar weights, and

$$d_G(s_0, \ldots, s_n) := \sum_{i=1}^{n} \Big(\underbrace{\|s_i - s_{i-1}\|_2}_{\text{Path length}} + \underbrace{K(s_i)}_{\text{Concavity weight}} \big(A_1 + A_2 \underbrace{|(s_i - s_{i-1}) \cdot v_1(s_i)|}_{\text{Path-component in direction of greatest concavity}}\big) \Big) \quad (2)$$

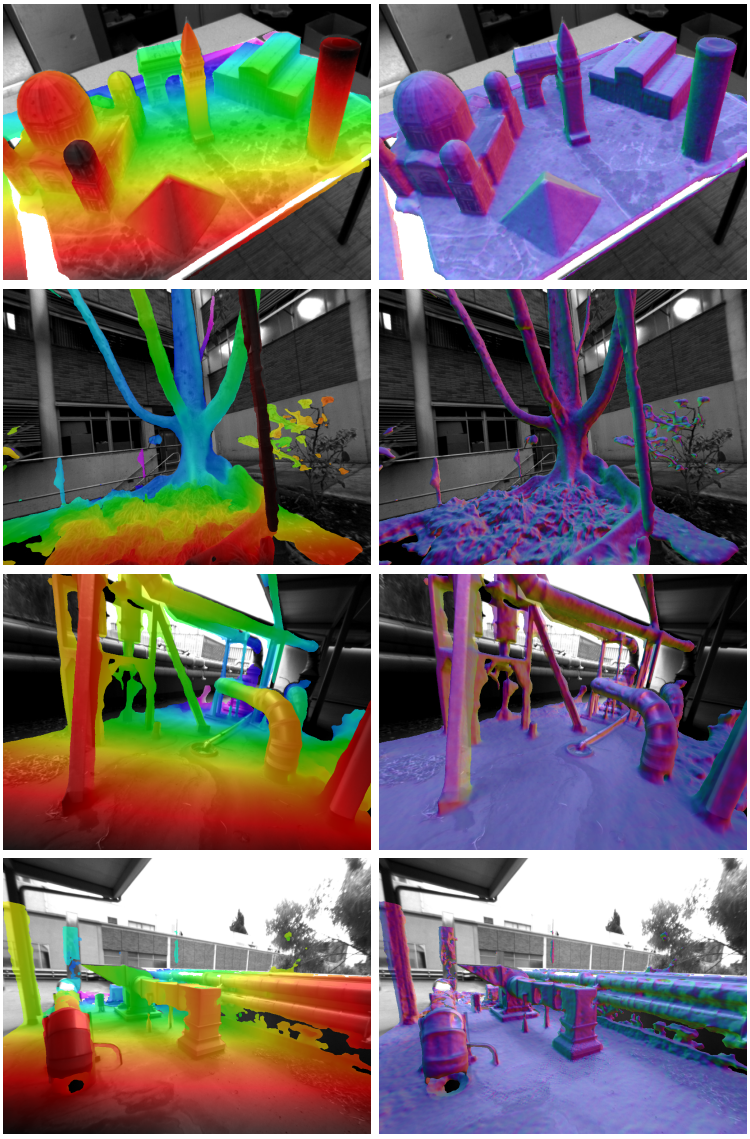

Figure 1: Dense reconstructions of indoor and outdoor scenes from monocular video: depth (left column) and normal (right column) maps. From top: *City of Sights*, *Tree*, *Industrial1*, *Industrial2* (described in section 3.2).

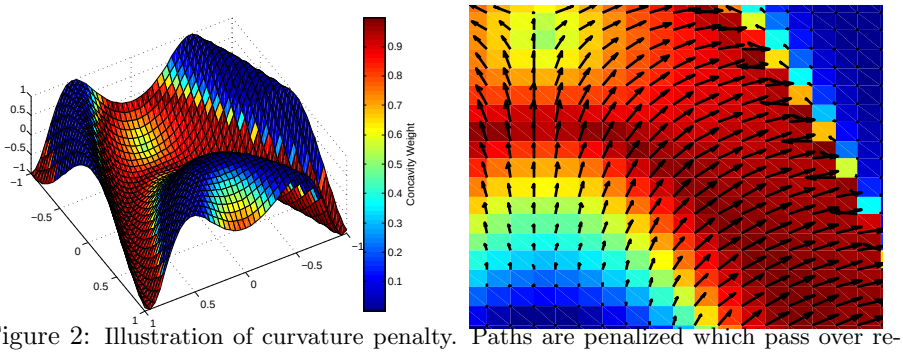

Figure 2: Illustration of curvature penalty. Paths are penalized which pass over regions with high concavity, especially in the direction of greatest concavity. At left, we show the scalar field $K(q) = \max\{0, k_1(q)^2/(k_1(q) + |k_2(q)|)\}$, on a sample curved surface, where $k_1(q)$ is the greatest positive principal curvature component, with associated vector field $v_1(q)$. At right, we show the weighted field $K(q)v_1(q)$ at each point.

A segmentation of the scene using these geometric homogeneity cues is a standard approach to 3D mesh segmentation, and is used as a baseline with which to evaluate our scale-aware segmentation in Sec. 3. A drawback of a purely geometric approach is that there is no unique scale appropriate for a task-relevant segmentation (such as segmenting potential objects for manipulation), as scenes typically consist of multiple scales of geometric primitives with strong violations of homogeneity between them (think of a coffee mug or a pineapple).

2.3 Constructing an Occlusion-Informed Geometry

To upgrade this curvature-augmented geometry to one capable of supporting queries beyond geometric homogeneity, trained detectors are typically used (e.g. **??**) to provide semantic homogeneity cues. In lieu of a battery of trained detectors for a fixed set of object classes, we obtain cues of topological characteristics *as seen by the viewer* through occlusions and use these to adapt the granularity of the distances on the scene to one relevant to the viewer's motion relative to the scene.

2.3.1 Single Image Occlusion-Based Segmentation

Salient occlusion boundaries provide a strong topological cue as to the arrangement of surfaces in the scene from a given vantage point and motion. Furthermore, they are derived from the measurements entirely at runtime and not dependent on prior training data. We implement the linear program formulation of **?**, which employs occlusion relationships between regions on the image plane as constraints on a depth-ordering of the image based on low level photometric or geometric homogeneity cues. Occlusion boundaries are obtained from salient depth discontinuities using the known geometry of the scene, and the segmentation is performed on a superpixelization of the image derived from the projected areas spanned by voxels associated to nodes of the scene mesh. These nodes are coarsified based on proximity to generate a computationally tractable number of superpixels which respect geometric boundaries in the images. Affinities between neighboring superpixels are found by computing the cost d_G between their

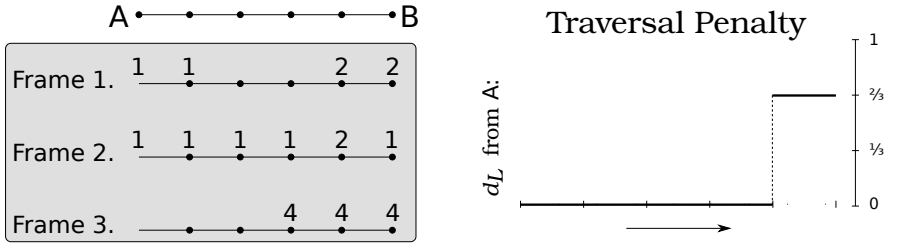

Figure 3: At left, example back-projected image segmentation labels c_t from three frames, over a sequence of nodes traversed from 'A' to 'B'. At right, the traversal penalty d_L accumulated over the traversal due to passing through nodes with conflicting image segmentation histories. The fact that some nodes are not visible in some frames means that penalties are not incurred along the same boundaries, depending on the direction of travel.

corresponding nodes. Sample single image segmentations are shown in section 3.2. Since the presence of salient occlusion cues is dependent on the viewpoint (and motion) of the camera, the back-projections of these segmentations give us a homogeneity cue *relevant to the viewer's motion* to combine with the geometric homogeneity cues of Sec. 2.2.2. To use these image segmentations c_t as topological homogeneity cues, we aggregate a history $C(q) = \{C_t(q)\}_{t=1}^{T}$ at each node q. If the node q is visible in the image at time t, then $C_t(q)$ takes the mode of assignments in c_t corresponding to the area its voxel subtends on the image plane. Zeroes in the history $C(q)$ denote frames in which the point p is not visible. A key assumption is that segmented regions in the images will be spatially consistent when back-projected onto the scene. If they consistently have disagreeing labels, then they were typically considered to occupy different depth-layers from the viewer's perspective and are likely not part of the same region. We quantify this by accumulating a penalty d_L along traversals of the scene that cross consistent image segmentation boundaries. This penalty is the normalized total number of frames for which the segmentation assignment changes, at least once, along the path (Eq. 3), as illustrated in Fig. 3.

$$d_L(s_0, \ldots, s_n) := \frac{1}{T}\underbrace{|\{t \ : \ \exists i, j \in 0, \ldots n, \ 0 \neq C_t(s_j) \neq C_t(s_i) \neq 0\}|}_{\text{Frames in which the layer assignment changes between } s_0 \text{ and } s_n} \tag{3}$$

Note that $0 \leq d_L \leq 1$.

2.3.2 Occlusion-Constrained Geometric Affinity

Secs. 2.2.2 and 2.3.1 present two different traversal costs along the nodes of the mesh. d_G models deviations from geometric homogeneity, and d_L models violations of image topology informed by occlusions. Nonparametric segmentation techniques (such as those used in Sec. 2.4) are preferred for generic segmentation tasks due to their ability to select the number of segments automatically. As a consequence, any combination of d_G and d_L between two nodes must change the structure of the resulting scene distance matrix at all scales in order to be effective. For example, a cost $d(q_i, q_j) = d_G(q_i, q_j) + d_L(q_i, q_j)$ will amplify geometric distances linearly when $d_L(q_i, q_j) \neq 0$, and have no impact on distances

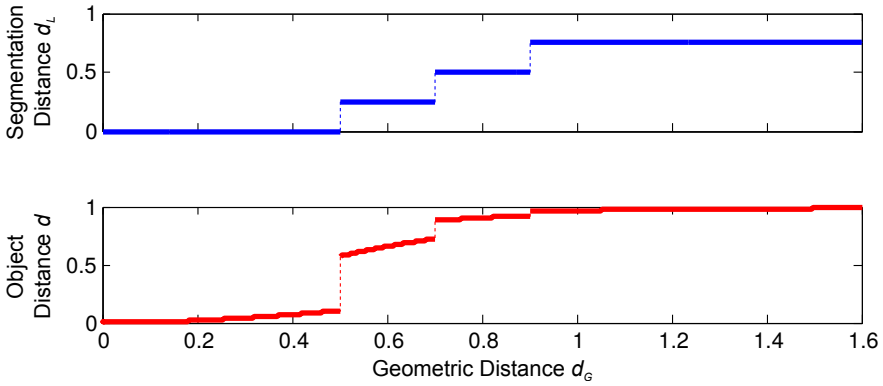

Figure 4: Example behavior of the combined traversal cost in Eq. 4 using artificial data. Note that as d_L increases, the rate of increase of d as a function of d_G accelerates up until saturation of the robust penalty.

within contiguous regions bounded by occlusions cues in the images. This will likely lead to over-segmentations of the scene in those regions when using non-parametric methods. Therefore a key design criterion for our adaptation of the geometric costs d_G between nodes using d_L is that they be *attenuated* when d_L is small and *amplified* when d_L is close to one.

To achieve this, and serve the dual purpose of providing a natural conversion from distances d_G to affinities for segmentation, we compute the cost of traversing a path between successive adjacent nodes on the scene using the Geman-McClure robust penalty (Eq. 4) with a scale parameter $\sigma_{\alpha,\varepsilon}(d_L)$ (Eq. 5).

$$d(q_i, q_j) = \min_{\left\{ \substack{s_0 \to \cdots \to s_n \\ q_i = s_0,\ q_j = s_n} \right\}} \left(1 + \frac{\sigma_{\alpha,\varepsilon}(d_L(s_0, \ldots, s_n))^2}{d_G(s_0, \ldots, s_n)^2} \right)^{-1}, \tag{4}$$

$\sigma_{\alpha,\varepsilon}(d_L)$ acts as a *scale shaping* function, the goal of which is to locally adapt the scale of the distance function based on the available evidence for object boundaries. If minimal evidence is present (d_L is small), $\sigma_{\alpha,\varepsilon}$ will be large and attenuate the increase in distance. If d_L is large and a consistent boundary in back-projected image segmentations is present then $\sigma_{\alpha,\varepsilon}(d_L)$ will shrink, accelerating the increase in distance. The parameters α and ε control the rate of decreasing scale and boundary values (at $d_L = 0$ and $d_L = 1$) respectively. Fig. 4 shows an example of the behavior of this choice of combined geodesic and adaptive scale using artificial data.

$$\sigma_{\alpha,\varepsilon}(d_L) = \frac{1 - \exp(-\alpha(1 - d_L) - \varepsilon)}{1 - \exp(-d_L - \varepsilon)} \tag{5}$$

The set of distances $d_i := \{d_{ij} \ : \ j \in S\}$ is computed in $O(|S|\, n_{\text{frames}})$ time, using a modified Dijkstra's algorithm on the nodes of the scene mesh.

2.4 Scene Segmentation

Discrete representations of complex scenes at high resolution can typically consist of many thousands of nodes, making both computation and storage of a

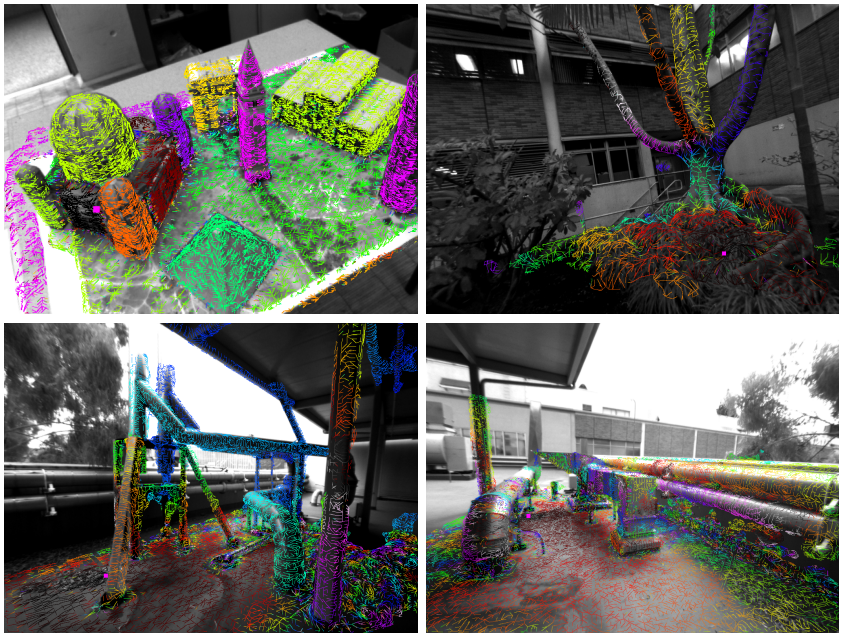

Figure 5: Sample geometric traversal costs d_G from a single point to all others across the scene overlaid on images. Colored lines indicate the path of the geodesic on the scene originating at the magenta square in each image, with distance increasing with changing colors, starting from zero (black).

pairwise distance matrix of geodesics between all nodes infeasible. To make segmentation based on pairwise distances tractable we generate a subgraph of the scene by uniformly sampling nodes subject to a minimum Euclidean distance and compute geodesics between them using the full resolution representation. To obtain a sparse segmentation of the scene, we apply a graph-based version of the DP-Means **?** algorithm, a low-variance asymptotic clustering algorithm derived from the Dirichlet process Gaussian mixture model **?**. The DP-Means algorithm was chosen for its nonparametric nature, i.e. its ability to select the number of objects in the scene automatically, and its computational speed. However, the original algorithm is only applicable to clustering data in a normed vector space; thus, we find an initial segmentation of the subgraph by globally optimizing a spectral relaxation **??** of the DP-Means cost, and refine the segmentation via kernelized **?** iterative updates. The partitioned subgraph is projected back to full mesh by performing a Voronoi tessellation of the scene discretization using the previously computed geodesics from each node in the subgraph.

A possible future extension of our segmentation pipeline is to enable adaptation to dynamically changing scales by treating more recent images segmentations preferentially, either through a fixed sliding window or decaying components of $d_L(s_0, \ldots, s_n)$. The Dynamic Means **?** algorithm, a low-variance asymptotic clustering algorithm based on the dependent Dirichlet process Gaussian mixture **?** can be used to make such a segmentation strategy temporally consistent. As in the batch case, this algorithm is ideal due its nonparametric ability to automatically discover the number of objects in the scene, and for its computational speed. However, Dynamic Means suffers from the same limitation as DP-Means; it is only applicable to data in a normed vector space. Therefore, for each video frame, we find an initial segmentation of the single frame alone using spectral clustering, and then enforce temporal consistency in the segmentation by using kernelized refinement iterations based on the Dynamic Means cost.

3 Evaluation

3.1 Comparison Methodology

We are not aware of benchmarks for evaluating scene segmentation inferred from monocular vision. While several RGB-D datasets for reconstruction and segmentation exist (such as **????**) they tend to have highly variable viewpoint scale, which makes the appropriate scale of a segmentation (based on occlusion cues) vary over time, in addition to having very similar scene content and geometry (indoor office and home scenes). Therefore, we have captured a set of diverse sequences, both indoor and outdoor, on which to test our dense reconstruction and segmentation pipeline, and included one provided by **?**. To generate ground truth for each sequence we manually segment the reconstruction into regions that correspond to objects at a scale appropriate for interaction from the video's perspective. As noted in section 2.1, objects compose a nested covering of sets, and the choice of which partition to use for groundtruth is subjective, though we have endeavored to select a fair partition into the dominant objects as visible from the videos. We compare the results of our occlusion-constrained segmentation to a baseline of segmentation using standard geometric homogene-

ity cues (described in section 2.2.2), for which typically a fixed scale parameter must be selected to perform segmentation, chosen to preserve as many of the dominant objects as possible without over-segmenting them. Note that it is infeasible for a single scale to accurately segment the entire scene, necessitating our adaptation of scale using occlusion cues from the viewer's motion. Baseline segmentations are compared numerically to object-level segmentations through F-score and precision-recall metrics. F-scores are computed following standard methodology **?**, to determine the agreement between ground-truth segments and computed segments.

Given a correspondence between computed cluster c_i, $i \in I$ and ground-truth regions g_j, $j \in J$, we compute F-scores as follows. Precision P_{ij} and recall R_{ij} are computed as the average (weighted by cluster size) ground-truth fraction of clusters, $P_{ij} = |c_i \cap g_j|/|c_i|$, which penalizes under-segmentation, and the fraction of the corresponding ground-truth region covered by a cluster, $R_{ij} = |c_i \cap g_j|/|g_j|$, which penalizes over-segmentation. A compromise measure $F_{ij} = 2P_{ij}R_{ij}/(P_{ij} + R_{ij})$ penalizes both. An optimal correspondence $\phi : I \to J$ is found by the Hungarian algorithm, maximizing the the total F-score,

$$F = \max_{\{\phi:I\to J\}} \frac{1}{|I|} \sum\nolimits_{i\in I} F_{i\phi(i)}. \tag{6}$$

A precision-recall curve may also be computed by comparing a thresholded affinity matrix $\delta_{M_{ij}>t}$ with the ground truth affinity matrix $\delta_{g_i=g_j}$. Since this is a monotonic function of distance along the scene, the precision-recall curve sampling over affinity thresholds allows us to evaluate the segmentation results across a range of scales.

3.2 Geometric and Occlusion-Constrained Segmentation Results

We present results in the form of re-projected segmentations and sample geodesics on four geometrically and topologically complex scenes, three of which were collected outdoors (Park, Industrial1, and Industrial2) and two of which contain notable multi-scale geometry (Park and City of Sights (CoS), made available by **?**). Please refer to our supplemental material for video results on these sequences.

Fig. 6 shows sample images and re-projected groundtruth segmentations for each scene, Fig. 7 shows sample occlusion-based image segmentation results, and Fig. 8 shows sample occlusion-constrained geodesics on the scene built using the occlusion-based image segmentations. Fig. 9 shows qualitative examples of our baseline geometric segmentation. Fig. 10 shows qualitative examples of our final scene segmentations, with numerical evaluations relative to groundtruth shown in Fig. 11.

In the CoS sequence the multi-scale geometry of the domed structure (a single object) makes the selection of a single scale infeasible, however the adaptive geodesic is able to shape distances on the scene based on the available image segmentations, enabling a correct segmentation. The Park sequence shows a scene with complex natural geometry on the ground and smoothly varying geometry on the nearby tree. Highly variable ground geometry makes the selection of a single scale unable to correctly segment both the smooth tree limbs (which occlude each other throughout the sequence) and the rough ground (as a single object).

These sequences demonstrate that adapting the geometry using occlusion-based image segmentation enables segmentation at the appropriate viewpoint scale for all dominant objects in the scene. Both Industrial sequences shows scenes of sophisticated topology and geometry, the segmentation of which is improved using our occlusion-informed geometry compared to the over-segmented results using standard geometric approaches.

The quantitative evaluation of Fig. 11 demonstrates a consistent improvement over a standard geometric segmentation when compared to our adaptive geodesic. The Precision-Recall curve comparisons serve to provide further support to the claim that no fixed scale treatment of purely geometric distances can correctly segment scenes with complex geometry, as varying affinity scale for both methods typically shows an increase in performance for our adaptive geodesic distances.

Fig. 12 shows a summary of timings for the various components of our system. Typical size of meshes used to represent the scene geometry are on the order of fifty thousand, with coarsified meshes for image segmentation on the order of two thousand, and subgraphs used for clustering on the order of several hundred. The entire system runs on a 3.5Ghz desktop machine (dense reconstruction, image, and scene segmentation).

4 Conclusions

We have presented a method to endow a scene, as densely reconstructed from monocular video, with a metric that incorporates geometric and topological information as seen by the viewer, as well as back-projected image statistics. While the latter are temporally inconsistent (they change with the video), the way they change is spatially consistent, an observation key to defining distances or affinities that allow us to partition the scene into coherent "objects". While one could employ trained object detectors at the outset to arrive at a semantic segmentation of the scene (and, by simple forward projection, of the video), we focus on low-level geometric and topological cues first, to segment the image into coherent regions, where one could then deploy object detectors if so desired. Semantic analysis of the scene involves object identities and relations, and knowledge of scene geometry and topology is key to infer the latter. This is our focus in this work.

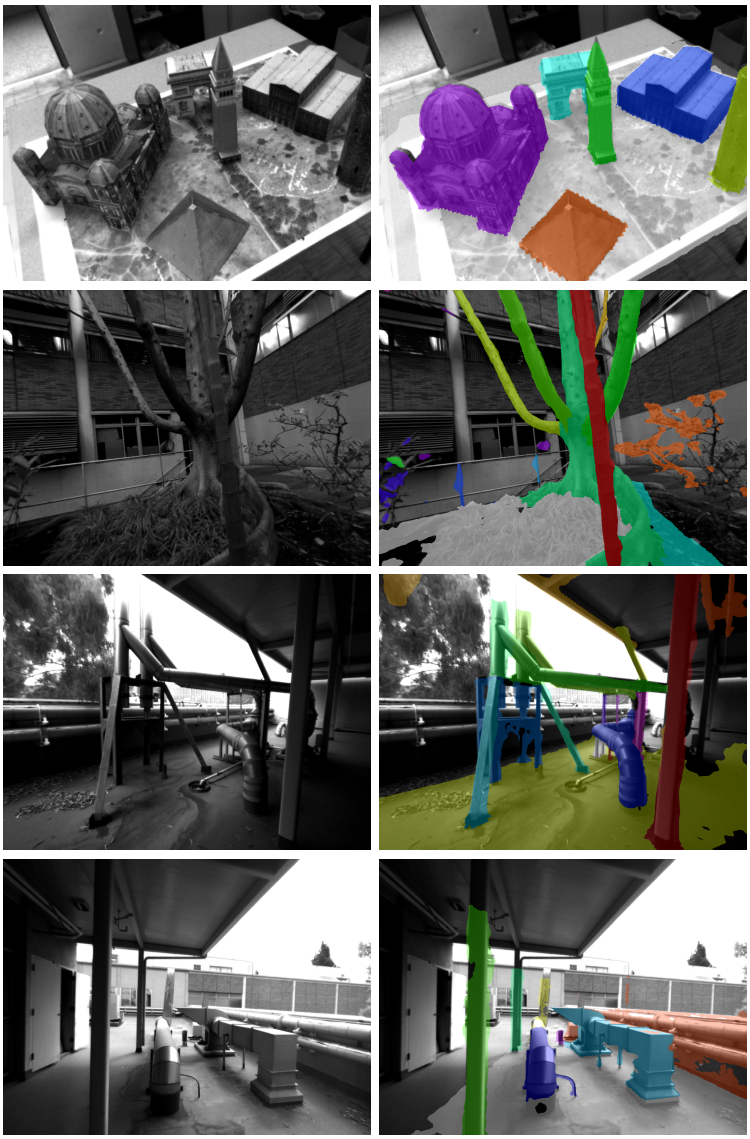

Figure 6: Sample frames (left) and re-projected groundtruth segmentations (right) for CoS (top), Park, Industrial1, Industrial2 (bottom). Different colors indicate different segments.

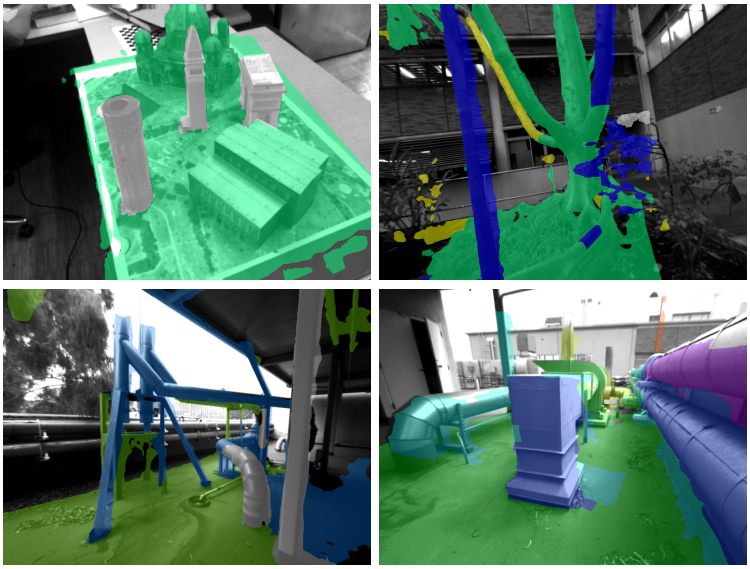

Figure 7: Sample single-image segmentation on frames from CoS (top left), Park (top right), Industrial1 (bottom left), Industrial2 (bottom right) inferred as described in Sec. 2.3.1. Different colors indicate different segments from occlusion-guided segmentation.

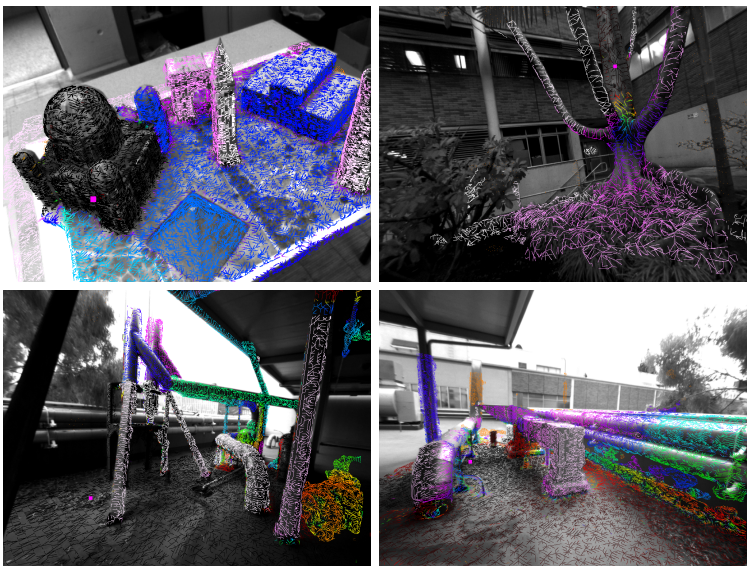

Figure 8: Sample occlusion-constrained geodesics on CoS (top left), Park (top right), Industrial1 (bottom left), Industrial2 (bottom right) built as described in Sec. 2.3.2. Colored lines indicate the path of the geodesic on the scene originating at the magenta square in each image, with distance coded between zero (black) and one (white) through intervening colors.

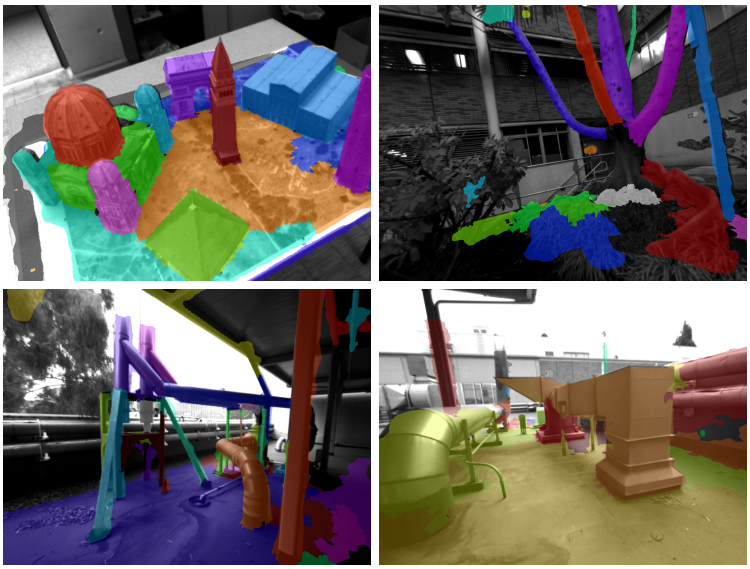

Figure 9: Sample re-projections of our baseline geometric segmentation on CoS (top left), Park (top right), Industrial1 (bottom left), Industrial2 (bottom right). Different colors indicate different segments.

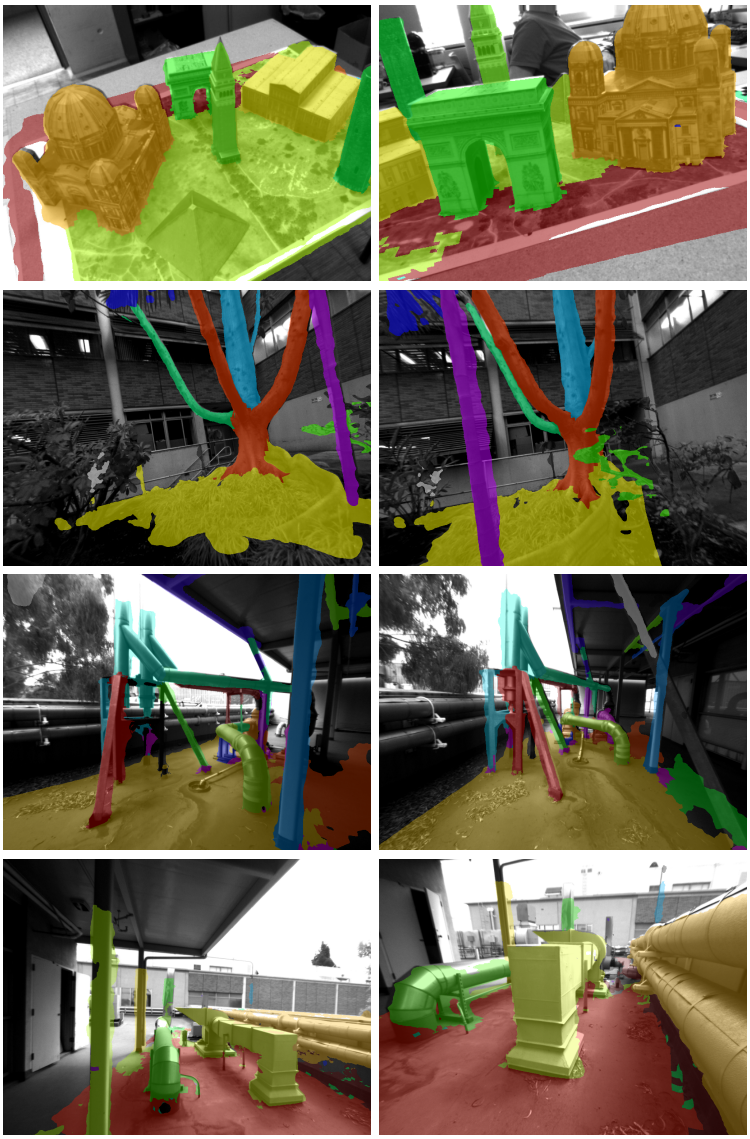

Figure 10: Sample re-projected segmentation results using our occlusion-constrained geodesics for CoS (top), Park, Industrial1, Industrial2 (bottom). Different colors indicate different segments.

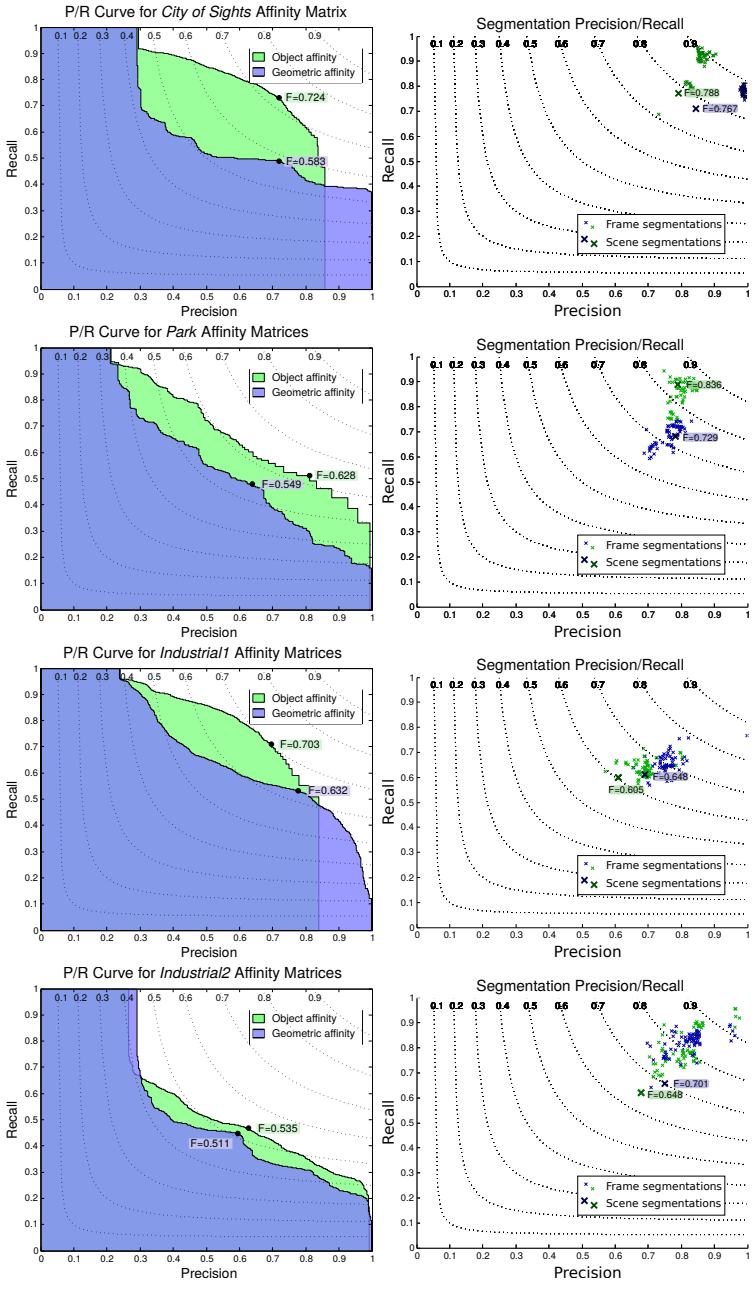

Figure 11: Quantitative evaluation of our segmentations vs. groundtruth. The curves at left show precision and recall scores induced by thresholding the affinity matrix at levels running from 0 to 1. Highlighted points indicate the best threshold for clustering, vis-à-vis the ground truth. At right, precision and recall are computed from final segmentations, as well as from re-projected image segmentations.

Component	Time	Hardware
1. Dense Reconstruction	30Hz Realtime	GPU
2. Image Segmentation	3s / frame	1 × CPU
3. Construct Mesh	.2s / frame	1 × CPU
4. Compute Geodesics	0.6s / traversal	N × CPU
5. Segmentation	1.2s	1 × CPU

Figure 12: Approximate timings on the CoS sequence with a final mesh size of roughly 57000 points. GPU is an Nvidia GTX 780. Note that traversing the mesh to compute geodesics for the sampled subgraph is parallelized. Typically a subgraph of several hundred nodes is used.