Multi-object shape estimation and tracking from silhouette cues

Li Guan, Jean-Sébastien Franco, Marc Pollefeys

Multi-Object Shape Estimation and Tracking from Silhouette Cues

Li Guan
UNC-Chapel Hill, U.S.A.
ETH-Zürich, Switzerland
lguan@cs.unc.edu

Jean-Sébastien Franco
LaBRI - INRIA Sud-Ouest
University of Bordeaux, France
jean-sebastien.franco@labri.fr

Marc Pollefeys
UNC-Chapel Hill, U.S.A.
ETH-Zürich, Switzerland
marc@cs.unc.edu

Abstract

This paper deals with the 3D shape estimation from silhouette cues of multiple moving objects in general indoor or outdoor 3D scenes with potential static obstacles, using multiple calibrated video streams. Most shape-from-silhouette techniques use a two-classification of space occupancy and silhouettes, based on image regions that match or disagree with a static background appearance model. Binary silhouette information becomes insufficient to unambiguously carve 3D space regions as the number and density of dynamic objects increases. In such difficult scenes, multi-view stereo methods suffer from visibility problems, and rely on color calibration procedures tedious to achieve outdoors. We propose a new algorithm to automatically detect and reconstruct scenes with a variable number of dynamic objects. Our formulation distinguishes between m different shapes in the scene by using automatically learnt view-specific appearance models, eliminating the color calibration requirement. Bayesian reasoning is then applied to solve the m-shape occupancy problem, with m updated as objects enter or leave the scene. Results show that this method yields multiple silhouette-based estimates that drastically improve scene reconstructions over traditional two-label silhouette scene analysis. This enables the method to also efficiently deal with multi-person tracking problems.

1. Introduction

Shape modeling from video is an important computer vision problem with numerous applications, such as 3D photography, virtual reality, 3D interaction or markerless motion capture. Silhouette-based techniques [13, 1] have been popularized thanks to their simplicity, speed, and general robustness to provide global shape and topology information about objects. Multi-view stereo techniques [12, 3, 17] prove more precise as they additionally recover object concavities, but are generally more computationally intense and require object appearance to be similar across views. The success of both families of approaches largely relies on the amount of control over the acquired scene, and is challenged in general, outdoor, densely populated scenes, where assumptions about visibility, lighting and scene content break. Primitive extraction and color calibration, both necessary for inter-view photocorrelation, become challenging or impossible. Binary silhouette reasoning with several objects is prone to large visual ambiguities, leading to misclassifications of significant portions of 3D space. Occlusion may occur between dynamic objects of interest. It can also be introduced by static objects in the scene, whose appearances are learned as part of the background model in many approaches, including ours. These occluders result in ambiguous and partial silhouette extractions.

In this paper we show that silhouette reasoning can be efficiently conducted by using distinct appearance models for objects, yielding a multi-silhouette modeling approach. We propose a Bayesian framework to merge silhouette cues arising from a set of dynamic objects, which accounts for all types of object occlusions and additional object localization constraints. This approach is shown to improve shape-from-silhouette estimation, can naturally be integrated with existing probabilistic occlusion inference methods, and can naturally benefit other vision problems such as multi-view tracking, segmentation, and general 3D modeling.

1.1. Previous work

Silhouette-based modeling in calibrated multi-view sequences has been largely popular, and yielded a large number of approaches to build volume-based [21] or surface-based [1] representations of the object's visual hull. The difficulty and focus in attention in modeling objects from silhouettes has gradually shifted from the pure 3D reconstruction issue to the sensitivity of visual hull representations to silhouette noise. In fully automatic modeling systems, silhouettes are usually extracted using background subtraction techniques [20, 4], which are difficult to apply outdoors and often locally fail due to changing lighting conditions, shadows, color space ambiguities, background object induced occlusion, among other causes. Several solutions have been proposed to address these problems, using a discrete optimization scheme [19], silhouette priors over multi-view sets [9], or silhouette cue integration using a sensor fusion paradigm [7]. Most existing reconstruction methods however focus on mono-object situations, and fail to address the specific multi-object issues of silhouette methods.

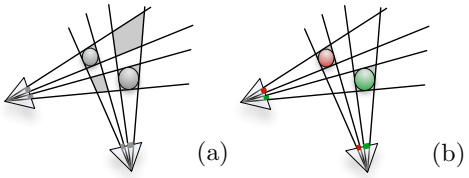

Figure 1. The principle of multi-object silhouette reasoning for shape modeling disambiguation. **Best viewed in color.**

While inclusive of the object's shape [13], visual hulls fail to capture object concavities but are usually very good at hinting toward the overall topology of a single observed object, a property that has been successfully used in a number of photometric-based methods to carve an initial silhouette-based volume [18, 8].

This ability to capture topologies breaks with the multiplicity of objects in the scene. In such cases 2-silhouettes are ambiguous in distinguishing between regions actually occupied by objects and unfortunate silhouette-consistent "ghost" regions. Such regions have been analyzed in the context of tracking applications to avoid committing to a "ghost" track [16]. The method we propose casts the problem of silhouette modeling at the multi-object level, where ghosts can naturally be eliminated based on per object silhouette consistency. Multi-object silhouette reasoning has been applied in the context of multi-object tracking [15, 6]. The reconstruction and occlusion problem has also been studied for the specific case of transparent objects [2]. Recent tracking efforts also use 2D probabilistic occlusion reasoning to improve object localization [11]. Static occluder analysis has also been proposed to analyze 3D scenes [10]. Our work is more general as it estimates full 3D shapes and copes with 3D occlusions both dynamic and static.

Perhaps the closest related work is the approach of Ziegler *et al.* [22], which builds 3D models deterministically from multiple label, user-provided silhouette segmentations. The approach we propose produces a more general probabilistic model that accounts for process noise and requires little or no user intervention.

1.2. Principle

The ghost phenomenon occurs when the configuration of the scene is such that regions of space occupied by objects of interest cannot be disambiguated from free-space regions that also happen to project inside all silhouettes, as the polygonal gray region in Fig. 1.2(a). Ghosts are increasingly likely as the number of observed objects rises, because it then becomes more difficult to find views that visually separate objects in the scene and carve out unoccupied regions of space. This problem is even aggravated for robust schemes, such as [7, 10], which do not strictly require silhouettes to be observed in every view. To address this problem, we initialize and learn a set of view-specific appearance models associated to m objects in the scene. The intuition is then that the probability of confusing ambiguous regions with real objects decreases, because the silhouette set corresponding to ghosts is then drawn from non object-consistent appearance model sets, as depicted in Fig. 1.2(b).

It is possible to process multiple silhouette labels in a deterministic, purely geometric fashion [22], but this comes at the expense of an arbitrary hard threshold for the number of views that define consistency. Silhouettes are then also assumed to be manually given and noiseless, which cannot be assumed for automatic processing. Using a volume representation of the 3D scene, we thus process multi-object sequences by examining each voxel in the scene using a Bayesian formulation (§2), which encodes the noisy causal relationship between the voxel and the pixels that observe it in a generative sensor model. In particular, given the knowledge that a voxel is occupied by a certain object among m possible in the scene, the sensor model explains what appearance distributions we are supposed to observe, corresponding to that object. It also encodes state information about the viewing line and potential obstructions from other objects, as well as a localization prior used to enforce the compactness of objects, which can be used to refine the estimate for a given instant of the sequence. Voxel sensor model semantics and simplifications are borrowed from the occupancy grid framework explored in the robotics community [5, 14]. The proposed method can also be seen as a multi-object generalization of previous probabilistic approaches focused on 2-label silhouette modeling [7, 10].

This scheme enables us to perform silhouette inference (§2.3) in a way that reinforces regions of space which are drawn from the same conjunction of color distributions, corresponding to one object, and penalizes appearance inconsistent regions, while accounting for object visibility. An algorithm (§3) is then proposed to integrate the inference framework in a fully automatic system. Because they are mutually dependent, specific steps are proposed for the problems of initialization, appearance model estimation, multi-object and occluder shape recovery.

2. Formulation

We consider a scene observed by n calibrated cameras. We assume a maximum of m dynamic objects of interest can be present in the scene. In this formulation we focus on the state of one voxel at position X chosen among the positions of the 3D lattice used to discretize the scene. We here model how knowledge about the occupancy state of voxel X influences image formation, assuming a static appearance model for the background has previously been observed. Because of occlusion relationships arising between objects, the zones of interest to infer the state of voxel X are its n viewing lines $\mathcal{L}_i$, $i \in \{1, \cdots, n\}$, with respect to the different views. In this paragraph we assume that some

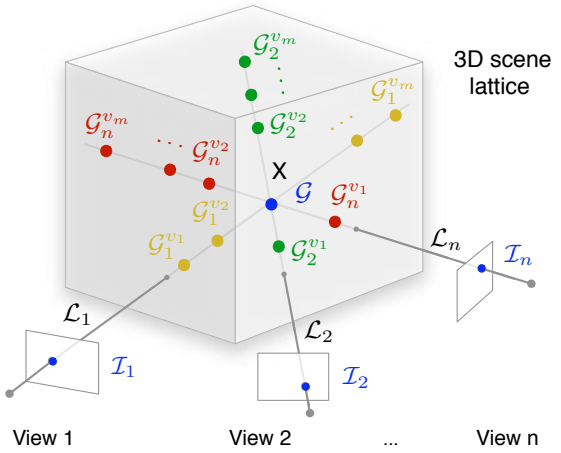

Figure 2. Overview of main statistical variables and geometry of the problem. $\mathcal{G}$ is the occupancy at voxel X and lives in a state space $\mathcal{L}$ of object labels. $\{\mathcal{I}_i\}$ are the color states observed at the n pixels where X projects. $\{\mathcal{G}_i^{v_j}\}$ are the states in $\mathcal{L}$ of the most likely obstructing voxels on the viewing line, for each of the m objects, enumerated in their order of visibility $\{v_j\}_i$.

prior knowledge about scene state is available for each voxel X in the lattice and can be used in the inference. Various uses of this assumption will be demonstrated in §3. A number of statistical variables are used to model the state of the scene, the image generation process and to infer $\mathcal{G}$, as depicted in figure Fig. 2.

2.1. Statistical Variables

Scene voxel state space. The occupancy state of X is represented by a variable $\mathcal{G}$. The particularity of our modeling lies in the multi-labeling characteristic of $\mathcal{G} \in \mathcal{L}$, where $\mathcal{L}$ is a set of labels $\{\emptyset, 1, \cdots, m, \mathcal{U}\}$. A voxel is either empty ($\emptyset$), one of m objects the model is keeping track of (numerical labels), or occupied by an unidentified object ($\mathcal{U}$). $\mathcal{U}$ is intended to act as a default label capturing all objects that are detected as different than background but not explicitly modeled by other labels, which proves useful for automatic detection of new objects (§3.3).

Observed appearance. The voxel X projects to a set of pixels, whose colors $\mathcal{I}_i$, $i \in 1, \cdots, n$ we observe in images. We assume these colors are drawn from a set of object and view specific color models whose parameters we note $\mathcal{C}_i^l$. More complex appearance models are possible using gradient or texture information, without loss of generality.

Latent viewing line variables. To account for inter-object occlusion, we need to model the contents of viewing lines and how it contributes to image formation. We assume some *a priori* knowledge about where objects lie in the scene. The presence of such objects can have an impact on the inference of $\mathcal{G}$ because of the visibility of objects and how they affect $\mathcal{G}$. Intuitively, conclusive information about $\mathcal{G}$ cannot be obtained from a view i if a voxel in front of $\mathcal{G}$ with respect to i is occupied by another object, for example. However, $\mathcal{G}$ directly influences the color observed if it is unoccluded and occupied by one of the objects. But if $\mathcal{G}$ is known to be empty, then the color observed at pixel $\mathcal{I}_i$ reflects the appearance of objects behind X in image i, if any. These visibility intuitions are modeled below (§2.2).

It is not meaningful to account for the combinatorial number of occupancy possibilities along the viewing rays of X. This is because neighboring voxel occupancies on the viewing line usually reflect the presence of the same object and are therefore correlated. In fact, assuming we witness no more than one instance of every one of the m objects along the viewing line, the fundamental information that is required to reason about X is the knowledge of presence and ordering of the objects along this line. To represent this knowledge, as depicted in Fig. 2, assuming prior information about occupancies is already available at each voxel, we extract, for each label $l \in \mathcal{L}$ and each viewing line $i \in \{1, \cdots, n\}$, the voxel whose probability of occupancy is dominant for that label on the viewing line. This corresponds to electing the voxels which best represent the m objects and have the most influence on the inference of $\mathcal{G}$. We then account for this knowledge in the problem of inferring X, by introducing a set of statistical occupancy variables $\mathcal{G}_i^l \in \mathcal{L}$, corresponding to these extracted voxels.

2.2. Dependencies Considered

We propose a set of simplifications in the joint probability distribution of the set of variables, that reflect the prior knowledge we have about the problem. To simplify the writing we will often note the conjunction of a set of variables as following: $\mathcal{G}_{1:n}^{1:m} = \{\mathcal{G}_i^l\}_{i\in\{1,\cdots,n\},l\in\{1,\cdots,m\}}$. We propose the following decomposition for the joint probability distribution $p(\mathcal{G}\ \mathcal{G}_{1:n}^{1:m}\ \mathcal{I}_{1:n}\ \mathcal{C}_{1:n}^{1:m})$:

$$p(\mathcal{G}) \prod_{l\in\mathcal{L}} p(\mathcal{C}_{1:n}^l) \prod_{i,l\in\mathcal{L}} p(\mathcal{G}_i^l|\mathcal{G}) \prod_i p(\mathcal{I}_i|\mathcal{G}\ \mathcal{G}_i^{1:m}\ \mathcal{C}_i^{1:m}) \quad (1)$$

Prior terms. $p(\mathcal{G})$ carries prior information about the current voxel. This prior can reflect different types of knowledge and constraints already acquired about $\mathcal{G}$, e.g. localization information to guide the inference (§3).

$p(\mathcal{C}_{1:n}^l)$ is the prior over the view-specific appearance models of a given object l. The prior, as written over the conjunction of these parameters, could express expected relationships between the appearance models of different views, even if not color-calibrated. Since the focus in this paper is on the learning of voxel X, we do not use this capability here and assume $p(\mathcal{C}_{1:n}^l)$ to be uniform.

Viewing line dependency terms. We have summarized the prior information along each viewing line using the m

voxels most representative of the m objects, so as to model inter-object occlusion phenomena. However when examining a particular label $\mathcal{G} = l$, keeping the occupancy information about $\mathcal{G}_i^l$ would lead us to account for intra-object occlusion phenomena, which in effect would lead the inference to favor mostly voxels from the front visible surface of the object l. Because we wish to model the *volume* of object l, we discard the influence of $\mathcal{G}_i^l$ when $\mathcal{G} = l$:

$$p(\mathcal{G}_i^k|\{\mathcal{G} = l\}) = \mathcal{P}(\mathcal{G}_i^k) \qquad \text{when } k \neq l \tag{2}$$

$$p(\mathcal{G}_i^l|\{\mathcal{G} = l\}) = \delta_\emptyset(\mathcal{G}_i^l) \qquad \forall l \in \mathcal{L}, \tag{3}$$

where $\mathcal{P}(\mathcal{G}_i^k)$ is a distribution reflecting the prior knowledge about $\mathcal{G}_i^k$, and $\delta_\emptyset(\mathcal{G}_i^k)$ is the distribution giving all the weight to label $\emptyset$. In (3) $p(\mathcal{G}_i^l|\{\mathcal{G} = l\})$ is thus enforced to be empty when $\mathcal{G}$ is known to be representing label l, which ensures that the same object is represented only once on the viewing line.

Image formation terms. The image formation term $p(\mathcal{I}_i|\mathcal{G}\ \mathcal{G}_i^{1:m}\ \mathcal{C}_i^{1:m})$ explains what color we expect to observe given the knowledge of viewing line states and per-object color models. We decompose each such term in two subterms, by introducing a local latent variable $\mathcal{S} \in \mathcal{L}$ representing the hidden silhouette state:

$$p(\mathcal{I}_i|\mathcal{G}\ \mathcal{G}_i^{1:m}\ \mathcal{C}_i^{1:m}) = \sum_{\mathcal{S}} p(\mathcal{I}_i|\mathcal{S}\ \mathcal{C}_i^{1:m}) p(\mathcal{S}|\mathcal{G}\ \mathcal{G}_i^{1:m}) \tag{4}$$

The term $p(\mathcal{I}_i|\mathcal{S}\ \mathcal{C}_i^{1:m})$ simply describes what color is likely to be observed in the image given the knowledge of the silhouette state and the appearance models corresponding to each object. $\mathcal{S}$ acts as a mixture label: if $\{\mathcal{S} = l\}$ then $\mathcal{I}_i$ is drawn from the color model $\mathcal{C}_i^l$. For objects ($l \in \{1, \cdots, m\}$) we typically use Gaussian Mixture Models (GMM) [20] to efficiently summarize the appearance information of dynamic object silhouettes. For background ($l = \emptyset$) we use per-pixel Gaussians as learned from pre-observed sequences, although other models are possible. When $l = \mathcal{U}$ the color is drawn from the uniform distribution, as we make no assumption about the color of previously unobserved objects.

Defining the silhouette formation term $p(\mathcal{S}|\mathcal{G}\ \mathcal{G}_i^{1:m})$ requires that the variables be considered in their visibility order, to model the occlusion possibilities. Note that this order can be different from $1, \cdots, m$. We note $\{\mathcal{G}_i^{v_j}\}_{j\in\{1,\cdots,m\}}$ the variables $\mathcal{G}_i^{1:m}$ as enumerated in the permutated order $\{v_j\}_i$ reflecting their visibility ordering on $\mathcal{L}_i$. If $\{g\}_i$ denotes the particular index after which the voxel X itself appears on $\mathcal{L}_i$, then we can re-write the silhouette formation term as $p(\mathcal{S}|\mathcal{G}_i^{v_1} \cdots \mathcal{G}_i^{v_g}\ \mathcal{G}\ \mathcal{G}_i^{v_{g+1}} \cdots \mathcal{G}_i^{v_m})$. A distribution of the following form can then be assigned to this term:

$$p(\mathcal{S}|\emptyset \cdots \emptyset\ l\ * \cdots *) = d_l(\mathcal{S}) \quad \text{with } l \neq \emptyset \tag{5}$$

$$p(\mathcal{S}|\emptyset\ \cdots\cdots\cdots\ \emptyset) = d_\emptyset(\mathcal{S}), \tag{6}$$

where $d_k(\mathcal{S}),\ k \in \mathcal{L}$ is a family of distributions giving strong weight to label k and lower equal weight to others, determined by a constant probability of detection $P_d \in [0, 1]$: $d_k(\mathcal{S} = k) = P_d$ and $d_k(\mathcal{S} \neq k) = \frac{1-P_d}{|\mathcal{L}|-1}$ to ensure summation to 1. (5) thus expresses that the silhouette pixel state reflects the state of the first visible non-empty voxel on the viewing line, regardless of the state of voxels behind it ("*"). (6) expresses the particular case where no occupied voxel lies on the viewing line, the only case where the state of $\mathcal{S}$ should be background: $d_\emptyset(\mathcal{S})$ ensures that $\mathcal{I}_i$ is mostly drawn from the background appearance model.

2.3. Inference

Estimating the occupancy at voxel X translates to estimating $p(\mathcal{G}|\mathcal{I}_{1:n}\ \mathcal{C}_{1:n}^{1:m})$ in Bayesian terms. We apply Bayes' rule using the joint probability distribution, marginalizing out the unobserved variables $\mathcal{G}_{1:n}^{1:m}$:

$$p(\mathcal{G}|\mathcal{I}_{1:n}\ \mathcal{C}_{1:n}^{1:m}) = \frac{1}{z} \sum_{\mathcal{G}_{1:n}^{1:m}} p(\mathcal{G}\ \mathcal{G}_{1:n}^{1:m}\ \mathcal{I}_{1:n}\ \mathcal{C}_{1:n}^{1:m}) \tag{7}$$

$$= \frac{1}{z} p(\mathcal{G}) \prod_{i=1}^{n} f_i^1 \tag{8}$$

$$\text{where } f_i^k = \sum_{\mathcal{G}_i^{v_k}} p(\mathcal{G}_i^{v_k}|\mathcal{G}) f_i^{k+1} \quad \text{for } k < m \tag{9}$$

$$\text{and } f_i^m = \sum_{\mathcal{G}_i^{v_m}} p(\mathcal{G}_i^{v_m}|\mathcal{G}) p(\mathcal{I}_i|\mathcal{G}\ \mathcal{G}_i^{1:m}\ \mathcal{C}_i^{1:m}) \tag{10}$$

The normalization constant z is easily obtained by ensuring summation to 1 of the distribution: $z = \sum_{\mathcal{G},\mathcal{G}_{1:n}^{1:m}} p(\mathcal{G}\ \mathcal{G}_{1:n}^{1:m}\ \mathcal{I}_{1:n}\ \mathcal{C}_{1:n}^{1:m})$. (7) is the direct application of Bayes rule, with the marginalization of latent variables. The sum in this form is intractable, thus we factorize the sum in (8). The sequence of m functions f_i^k specify how to recursively compute the marginalization with the sums of individual $\mathcal{G}_i^k$ variables appropriately subsumed, so as to factor out terms not required at each level of the sum. Because of the particular form of silhouette terms in (5), this sum can be efficiently computed by noting that all terms after a first occupied voxel of the same visibility rank k share a term of identical value in $p(\mathcal{I}_i|\emptyset \cdots \emptyset\ \{\mathcal{G}_i^{v_k} = l\}\ * \cdots *) = \mathcal{P}_l(\mathcal{I}_i)$. They can be factored out of the remaining sum, which sums to 1 being a sum of terms of a probability distribution, leading to the following simplification of (9), $\forall k \in \{1, \cdots, m-1\}$:

$$f_i^k = p(\mathcal{G}_i^{v_k} = \emptyset|\mathcal{G}) f_i^{k+1} + \sum_{l\neq\emptyset} p(\mathcal{G}_i^{v_k} = l|\mathcal{G}) \mathcal{P}_l(\mathcal{I}_i) \tag{11}$$

3. 3D Modeling and Localization Algorithm

We have presented in §2 a generic framework to infer the occupancy probability of a voxel X and thus deduce how

likely it is for X to belong to one of m objects. Some additional work is required to use it to model objects in practice. The formulation explains how to compute the occupancy of X if some occupancy information about the viewing lines is already known. Thus the algorithm needs to be initialized with a coarse shape estimate, whose computation is discussed in §3.1. Intuitively, object shape estimation and tracking are complementary and mutually helpful tasks. We explain in §3.2 how object localization information is computed and used in the modeling. To be fully automatic, our method uses the inference label $\mathcal{U}$ to detect objects not yet assigned to a given label and learn their appearance models (§3.3). Finally, it has been shown that static occluders can be computed using silhouette occlusion reasoning [10]. This reasoning can easily be integrated in our approach and help the inference be robust to static occluders (§3.4). The algorithm at every time instance is summarized in Alg. 1.

Algorithm 1: Dynamic Scene Reconstruction

Input: Frames at a new time instance for all views
Output: 3D object shapes in the scene

Coarse Inference;
if *new object enters the scene* **then**
    add a label for the new object;
    initialize foreground appearance model;
    go to step 1;
Refined Inference;
static occluder inference;
update object location and prior;
return

3.1. Shape Initialization and Refinement

The proposed formulation relies on some available prior knowledge about the scene occupancies and dynamic object ordering. Thus part of the occupancy problem must be solved to bootstrap the algorithm. Fortunately, using multi-label silhouette inference with no prior knowledge about occupancies or consideration for inter-object occlusions provides a decent initial m-occupancy estimate. This simpler inference case can easily be formulated by simplifying occlusion related variables from (8):

$$p(\mathcal{G}|\mathcal{I}_{1:n}\,\mathcal{C}_{1:n}^{1:m}) = \frac{1}{z}p(\mathcal{G})\prod_{i=1}^{n}p(\mathcal{I}_i|\mathcal{G}\,\mathcal{C}_i^{1:m}) \qquad (12)$$

This initial *coarse inference* can then be used to infer a second, *refined inference*, this time accounting for viewing line obstructions, given the voxel priors $p(\mathcal{G})$ and $\mathcal{P}(\mathcal{G}_i^j)$ of equation (2) computed from the coarse inference. The prior over $p(\mathcal{G})$ is then used to introduce soft constraints to the inference. This is possible by using the coarse inference result as the input of a simple localization scheme, and using the localization information in $p(\mathcal{G})$ to enforce a compactness prior over the m objects, as discussed in §3.2.

3.2. Object Localization

We use a localization prior to enforce the compactness of objects in the inference steps. For the particular case where walking people represent the dynamic objects, we take advantage of the underlying structure of the dataset, by projecting the maximum probability over a vertical voxel column on the horizontal reference plane. We then localize the most likely position of objects by sliding a fixed-size window over the resulting 2D probability map for each object. The resulting center is subsequently used to initialize $p(\mathcal{G})$, using a cylindrical spatial prior. This favors objects localized in one and only one portion of the scene and is intended as a soft guide to the inference. Although simple, this tracking scheme is shown to outperform state of the art methods (§4.2), thanks to the rich shape and occlusion information modelled.

3.3. Automatic Detection of New Objects

The main information about objects used by the proposed method is their set of appearances in the different views. These sets can be learned offline by segmenting each observed object alone in a clear, uncluttered scene before processing multi-objects scenes. More generally, we can initialize object color models in the scene automatically. To detect new objects we compute $\mathcal{U}$'s object location and volume size during the coarse inference, and track the unknown volume just like other objects as described in §3.2. A new dynamic object inference label is created (and m incremented), if all of the following criteria are satisfied:

- The entrance is only at the scene boundaries
- $\mathcal{U}$'s volume size is larger than a threshold
- Subsequent updates of $\mathcal{U}$'s track are bounded

To build the color model of the new object, we project the maximum voxel probability along the viewing ray to the camera view, threshold the image to form a "silhouette mask", and choose pixels within the mask as training samples for a GMM appearance model. Samples are only collected from unoccluded silhouette portions of the object, which can be verified from the inference. Because the cameras may be badly color-calibrated, we propose to train an appearance model for each camera view separately. This approach is fully evaluated in §4.1.

3.4. Occluder computation

The existing algorithm in [10] computes dynamic object binary occupancy distributions at every voxel. It then analyzes the presence of dynamic object dominant probabilities of occupancy in front and behind of the voxel on its viewing lines, for every view and passed time instant of the sequence. Such dominant occupancies are then used to accumulate cues about occluder occupancy at the current inferred voxel. The same formulation can easily be used and

extended using the analysis presented in this paper. At every time instant the dominant occupancy probabilities of m objects are already extracted; the two dominant occupancies in front and behind the current voxel X can be used in the occupancy inference formulation of [10]. The occlusion occupancy inference then benefits from the disambiguation inherent to multi-silhouette reasoning.

4. Results and Evaluations

We have used four multi-view sequences to validate our approach. Eight 30Hz 720×480 DV cameras surrounding the scene in a semi-circle were used for the CLUSTER and BENCH sequences. The LAB and SCULPTURE sequences are provided by [11] and [10] respectively for comparison.

	Cam. No.	Dynamic Obj. No.	Occluder
CLUSTER (outdoor)	8	5	no
BENCH (outdoor)	8	0 - 3	yes
LAB (indoor)	15	4	no
SCULPTURE (outdoor)	9	2	yes

Cameras in each data sequence are geometrically calibrated but not color calibrated. The background model is learned per-view using a single Gaussian color model at every pixel, with training images. Although simple, the model proves sufficient, even in outdoor sequences subject to background motion, foreground object shadows, window reflections and substantial illumination changes, showing the robustness of the method to difficult real conditions.

For dynamic object appearance models of the CLUSTER, LAB and SCULPTURE data sets, we train a RGB GMM model for each person in each view with manually segmented foreground images. This is done offline. For the BENCH sequence however, appearance models are initialized online automatically.

The time complexity is $\mathcal{O}(nmV)$, with n the number of cameras, m the number of objects in the scene, and V the scene volume resolution. We process the data sets on a 2.4 GHz Core Quad PC with computation times varying of 1-4 min per time step. The very strong locality inherent to the algorithm and preliminary benchmarks suggest that around 10 times faster performance could be achieved using a GPU implementation. **Please refer to the supplemental videos for complete results.**

4.1. Appearance Modeling Validation

It is extremely hard to color-calibrate a large number of cameras, not to mention under varying lighting conditions, as in a natural outdoor environment. To show this, we compare different appearance modeling schemes in Fig. 3, for a frame of the outdoor BENCH dataset. Without loss of generality, we use GMMs. The first two rows compare silhouette extraction probabilities using the color models

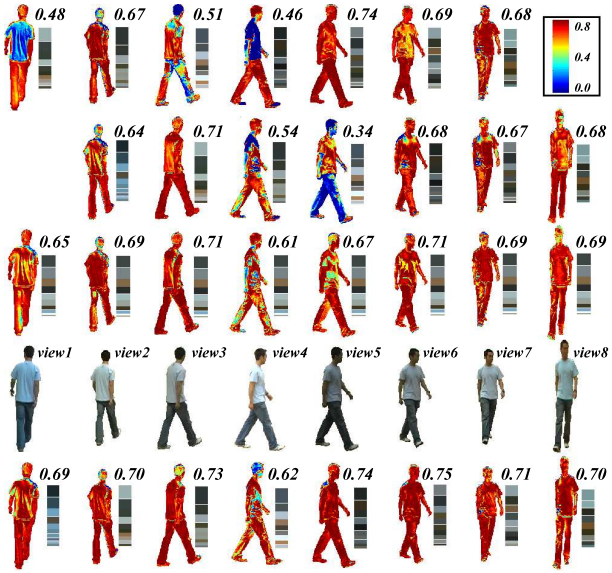

Figure 3. Appearance model analysis. A person in eight views is displayed in row 4. A GMM model $\mathcal{C}_i$ is trained for view $i \in [1, 8]$. A global GMM model $\mathcal{C}_0$ over all views is also trained. Row 1, 2, 3 and 5 compute $\mathcal{P}(\mathcal{S}|\mathcal{I}, \mathcal{B}, \mathcal{C}_{i+1})$, $\mathcal{P}(\mathcal{S}|\mathcal{I}, \mathcal{B}, \mathcal{C}_{i-1})$, $\mathcal{P}(\mathcal{S}|\mathcal{I}, \mathcal{B}, \mathcal{C}_0)$ and $\mathcal{P}(\mathcal{S}|\mathcal{I}, \mathcal{B}, \mathcal{C}_i)$ for view i respectively, with $\mathcal{S}$ the foreground label, $\mathcal{I}$ the pixel color, $\mathcal{B}$ the uniform background model. The probability is displayed according to the color scheme at the top right corner. The average probability over all pixels in the silhouette region and the mean color modes of the applied GMM model are shown for each figure. **Best viewed in color.**

of spatially neighboring views. These indicate that stereo approaches which heavily depend on color correspondence between neighboring views are very likely to fail in the natural scenarios, especially when the cameras have dramatic color variations, such as in view 4 and 5. The global appearance model on row 3 performs better than row 1 and 2, but this is mainly due to its compensation between large color variations across camera views, which at the same time, decreases the model's discriminability. The last row obviously is the winner where a color appearance model is independently maintained for every camera view. We hereby use the last scheme in our system. Once the model is trained, we do not update it as time goes by, which could be an easy extension for robustness.

4.2. Densely Populated Scene

The CLUSTER sequence is a particularly challenging configuration: five people are on a circle of less than 3m. in diameter, yielding an extremely ambiguous and occluded situation at the circle center. Despite the fact that none of them are being observed in all views, we are still able to recover the people's label and shape. Images and results are shown in Fig. 4. The naive 2-label reconstruction (probabilistic visual hull) yields large volumes with little separation between objects, because the entire scene configuration

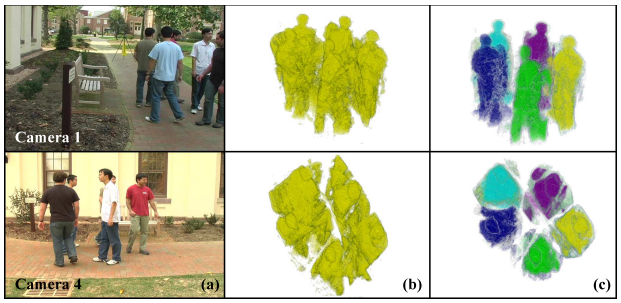

Figure 4. Result from 8-view CLUSTER dataset. (a) Two views at frame 0. (b) Respective 2-labeled reconstruction. (c) More accurate shape estimation using our algorithm. **Best viewed in color.**

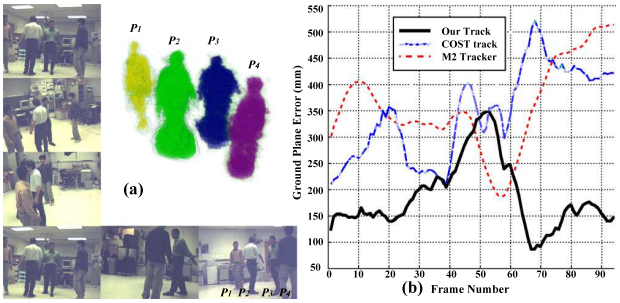

Figure 5. LAB dataset result from [11]. (a) 3D reconstruction with 15 views at frame 199 (b) 8-view tracking result comparison with methods in [11], [15] and the ground truth data. Mean error in ground plane estimate in mm is plotted. **Best viewed in color.**

is too ambiguous. Adding tracking prior information estimates the most probable compact regions and eliminates large errors, at the expense of dilation and lower precision. Accounting for viewing line occlusions enables the model to recover more detailed information, such as the limbs.

The LAB sequence [11] with poor image contrast is also processed. The reconstruction result from all 15 cameras is shown in Fig. 5. Moreover, in order to evaluate our localization prior estimation, we compare our tracking method (§3.2) with the ground truth data, the result of [11] and [15]. We use the same 8 cameras as in [15] for the comparison, shown in Fig. 5(b). Although slower in its current implementation (2 min. per time step) our method is generally more robust in tracking, and also builds 3D shape information. Most existing tracking methods only focus on a tracking envelope and do not compute precise 3D shapes. This shape information is what enables our method to achieve comparable or better precision.

4.3. Automatic Appearance Model Initialization

The automatic dynamic object appearance model initialization has been tested using the BENCH sequence. Three people are walking into the empty scene one after another. By examining the unidentified label $\mathcal{U}$, object appearance models are initialized and used for shape estimation in subsequent frames. Volume size evolution of all labels are shown in Fig. 6 and the reconstructions at two time instants are shown in Fig. 7.

During the sequence, $\mathcal{U}$ has three major volume peaks due to three new persons entering the scene. Some smaller perturbations are due to shadows on the bench or the ground. Besides automatic object appearance model initialization, the system robustly re-detects and tracks the person who leaves and re-enters the scene. This is because once the label is initialized, it is evaluated for every time instant, even if the person is out of the scene. The algorithm can easily be improved to handle leaving/reentering labels transparently.

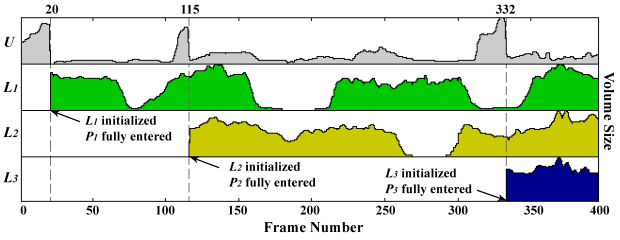

Figure 6. Appearance model automatic initialization with the BENCH sequence. The volume of $\mathcal{U}$ increases if a new person enters the scene. When an appearance model is learned, a new label is initialized. During the sequence, L_1 and L_2 volumes drop to near zero because they walk out of the scene on those occasions.

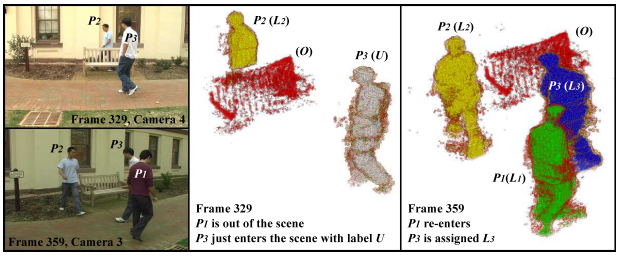

Figure 7. BENCH result. Person numbers are assigned according to the order their appearance models are initialized. At frame 329, P_3 is entering the scene. Since it's P_3's first time into the scene, he is captured by label $\mathcal{U}$ (gray color). P_1 is out of the scene at the moment. At frame 359, P_1 has re-entered the scene. P_3 has its GMM model already trained and label L_3 assigned. The bench as a static occluder is being recovered. **Best viewed in color.**

4.4. Dynamic Object & Occluder Inference

The BENCH sequence demonstrates the power of our automatic appearance model initialization as well as the integrated occluder inference of the *"bench"* as shown in Fig. 7 between frame 329 and 359. Check Fig. 6 about the scene configuration during that period. The complete sequence is also given in the supplemental video.

We also compute result for SCULPTURE sequence from [10] with two persons walking in the scene, as shown in Fig. 8. For the dynamic objects, we manage to get much

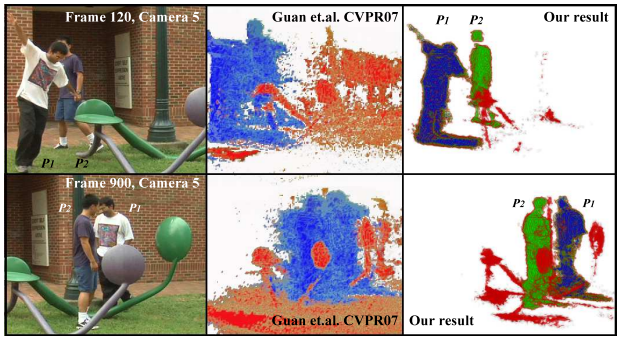

Figure 8. SCULPTURE data set comparison. While both [10] and our method recover the static sculpture, our method resolves inter-occlusion ambiguities, and estimates much better dynamic object shapes. **Best viewed in color.**

cleaner shapes when the two persons are close to each other, and more detailed shapes such as extended arms. For the occluder, we are able to recover the fine shape too, while [10] has a lot of noise, due to the occluder inference using ambiguous regions when people are clustered.

5. Discussion

We have proposed a Bayesian method to build 3D shapes from multi-object silhouette cues. The appearances of objects are used to disambiguate free regions of space which project inside silhouettes, and occlusion information and object localization priors are used to update the representation iteratively so as to refine the resulting shapes. Our results show that the shapes obtained using this approach yield significantly better results than pure silhouette reasoning, which makes no distinction between different objects. This new multi-silhouette inference algorithm is robust to very difficult conditions, and can prove very useful for various vision tasks such as tracking, localization and 3D reconstruction, in highly cluttered scenes with densely packed dynamic object groups. A large number of extensions can be tested on the basis of the framework provided, including more general and complex appearance modeling, different enforcements of the compactness of objects, a more general management of objects entering and leaving the scene. It is possible to analyze object label transition, for example a static object in the scene might be moved to a different place, and a person might come and sit statically on the bench. Temporal consistency constraints could also be included in stronger forms, to enforce temporal continuity of the reconstruction and smoothness of the flow in the scene.

Acknowledgments: We would like to thank A. Gupta *et.al.* [11, 15] for providing us the 16-camera dataset. This work was partially supported by David and Lucille Packard Foundation Fellowship, and NSF Career award IIS-0237533.

References

[1] B. G. Baumgart. *Geometric Modeling for Computer Vision*. PhD thesis, CS Dept, Stanford U., Oct. 1974.

[2] J. S. Bonet and P. Viola. Roxels: Responsibility weighted 3d volume reconstruction. In *ICCV*, vol. I, p. 418–425, 1999.

[3] A. Broadhurst, T. Drummond, and R. Cipolla. A probabilistic framework for the Space Carving algorithm In *ICCV'01*, p. 388–393, 2001.

[4] G. Cheung, T. Kanade, J.-Y. Bouguet, and M. Holler. A real time system for robust 3d voxel reconstruction of human motions. In *CVPR*, II:714 – 720, 2000.

[5] A. Elfes. Using occupancy grids for mobile robot perception and navigation. *IEEE Computer, Special Issue on Autonomous Intelligent Machines*, 22(6):46–57, June 1989.

[6] F. Fleuret, J. Berclaz, R. Lengagne, and P. Fua. Multi-camera people tracking with a probabilistic occupancy map. *PAMI*, 2007.

[7] J.-S. Franco and E. Boyer. Fusion of multi-view silhouette cues using a space occupancy grid. *ICCV'05*, II:1747–1753.

[8] Y. Furukawa and J. Ponce. Carved visual hulls for image-based modeling. In *ECCV*, 2006.

[9] K. Grauman, G. Shakhnarovich, and T. Darrell. A bayesian approach to image-based visual hull reconstruction. In *CVPR*, vol. I, p. 187–194, 2003.

[10] L. Guan, J.-S. Franco, and M. Pollefeys. 3D Occlusion Inference from Silhouette Cues. In *CVPR*, 2007.

[11] A. Gupta, A. Mittal, and L. S. Davis. Cost: An approach for camera selection and multi-object inference ordering in dynamic scenes. In *ICCV*, 2007.

[12] K. Kutulakos, and S. Seitz. A Theory of Shape by Space Carving. In *IJCV*, 2000.

[13] A. Laurentini. The Visual Hull Concept for Silhouette-Based Image Understanding. *PAMI*, 16(2):150–162, 1994.

[14] D. Margaritis and S. Thrun. Learning to locate an object in 3d space from a sequence of camera images. In *ICML'98*.

[15] A. Mittal and L. S. Davis. M2tracker: A multi-view approach to segmenting and tracking people in a cluttered scene. *IJCV*, 51(3):189–203, February 2003.

[16] K. Otsuka and N. Mukawa. Multiview occlusion analysis for tracking densely populated objects based on 2-d visual angles. In *CVPR*, I:90–97, 2004.

[17] S. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *CVPR*, 2006.

[18] S. N. Sinha and M. Pollefeys. Multi-view reconstruction using photo-consistency and exact silhouette constraints: A maximum-flow formulation. In *ICCV*, 2005.

[19] D. Snow, P. Viola, and R. Zabih. Exact voxel occupancy with graph cuts. In *CVPR*, p. 345–353, 2000.

[20] C. Stauffer and W. E. L. Grimson. Adaptive background mixture models for real-time tracking. *CVPR'99*, II:246–252.

[21] R. Szeliski, D. Tonnesen, and D. Terzopoulos. Modeling Surfaces of Arbitrary Topology with Dynamic Particles. In *CVPR*, p. 82–87, 1993.

[22] R. Ziegler, W. Matusik, H. Pfister, and L. McMillan. 3d reconstruction using labeled image regions. In *EG symposium on Geometry processing*, p. 248–259, 2003.