Multi-view occlusion reasoning for probabilistic silhouette-based dynamic scene reconstruction

Li Guan, Jean-Sébastien Franco, Marc Pollefeys

Editorial Manager(tm) for International Journal of Computer Vision
Manuscript Draft

Manuscript Number:

Title: Probabilistic Multi-view Dynamic Scene Reconstruction and Occlusion Reasoning from Silhouette Cues

Article Type: S.I.: Prob. Models for Image Under.

Section/Category:

Keywords: multi-view 3D reconstruction; probability; graphical model; Bayes rule; occluder

Corresponding Author: Mr. Li Guan, M.Sc

Corresponding Author's Institution: University of North Carolina-Chapel Hill

First Author: Li Guan, M.Sc

Order of Authors: Li Guan, M.Sc; Jean-Sebastien Franco, Ph.D.; Marc Pollefeys, Ph.D.

Manuscript Region of Origin:

Abstract: In this paper, we present a probabilistic multi-view algorithm to estimate object shapes in a 3D dynamic scene using their silhouette cues. We assign every object a distinctive label. Each label is associated with automatically learnt view-specific appearance models of the respective object to bypass the photometric calibration of the system. We also introduce generative probabilistic sensor models, and analyze the graphical dependencies between the sensor observations and object labels. Bayesian reasoning is then applied to achieve robust reconstruction against real-world environment challenges, such as lighting variations and occlusion. One of our main contributions is to explicitly account for visual occlusions: (1) Static occluders can be automatically detected and their 3D shapes are fully recovered as a byproduct of inference; (2) Ambiguities due to inter-occlusion between the dynamic objects can be alleviated, and the final

reconstruction quality is drastically improved. Several real-world datasets are tested to demonstrate the power of this framework.

International Journal of Computer Vision manuscript No.
(will be inserted by the editor)

Probabilistic Multi-view Dynamic Scene Reconstruction and Occlusion Reasoning from Silhouette Cues

Li Guan · Jean-Sébastien Franco · Marc Pollefeys

Received: date / Accepted: date

Abstract In this paper, we present a probabilistic multi-view algorithm to estimate object shapes in a 3D dynamic scene using their silhouette cues. We assign every object a distinctive label. Each label is associated with automatically learnt view-specific appearance models of the respective object to bypass the photometric calibration of the system. We also introduce generative probabilistic sensor models, and analyze the graphical dependencies between the sensor observations and object labels. Bayesian reasoning is then applied to achieve robust reconstruction against real-world environment challenges, such as lighting variations and occlusion. One of our main contributions is to explicitly account for visual occlusions: (1) Static occluders can be automatically detected and their 3D shapes are fully recovered as a byproduct of inference; (2) Ambiguities due to inter-occlusion between the dynamic objects can be alleviated, and the final reconstruction quality is drastically improved. Several real-world datasets are tested to demonstrate the power of this framework.

Keywords multi-view 3D reconstruction · probability · graphical model · Bayes rule · occluder

Li Guan
UNC-Chapel Hill
Tel.: +1-919-962-1771
Fax: +1-919-962-1799
E-mail: lguan@cs.unc.edu

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

Marc Pollefeys
ETH-Zürich, Switzerland
UNC-Chapel Hill, USA
E-mail: marc.pollefeys@inf.ethz.ch; marc@cs.unc.edu

1 Introduction

3D shape reconstruction from real world imagery is an important research topic in computer vision. In this paper, we focus on the problem of recovering a time varying dynamic scene involving moving and/or stationary objects from multiple video streams with fixed and known geometric poses. The choice of a multi-view solution is a must, because with dynamic objects in the scene, it is generally impossible for a single camera to get sufficient 3D information of an object at any time instant. This setup has many applications such as video games, animation, 3D TV, virtual reality, medical surgery, architectural design, performance training, digital documentary, etc.

Two well-known categories of the multi-view reconstruction algorithms are - (1) Shape from Photo-consistency/multi-view stereo approaches [24,4,32,35,33], which recover the dense correspondence across views using the appearance-consistency constraint. Their recovered shapes are proven to be precise as certain object concavities are recovered. But these methods are generally computationally intense and require object appearance to be similar across views. (2) Shape from Silhouette techniques [28,29,26,11], which generally assume the foreground object silhouette in an image can be separated from the background. Along with the camera viewing parameters, the back-projected silhouette cones intersect to form the visual hull [2,25], an approximate shape of the original object. Since silhouette-based algorithms are relatively simple, fast, and output the global shape and topology information of an object, they are good choices for dynamic scene analysis. A more important advantage of the silhouette-based methods is that they do not require object appearance to be similar across views. The generally tedious camera network radiometric calibration is thus not required. In some cases, this is critical, for example, in a natural outdoor environment with varying sunlight, the calibra-

tion is extremely difficult, because the constant illumination assumption which is required for most of the state-of-the-art approaches [20,21,39], does not hold anymore. Therefore, in this paper, we focus on exploring silhouette cues only.

Silhouette-based methods have their own difficulties: the silhouette computation is highly dependent on the per-view appearance-based background modeling, which is usually sensitive to imaging sensor noise, shadows and illumination variations in the scene. Also it is ambiguous when the modeled object has a similar appearance as the background. Therefore, Shape from Silhouette methods are usually used in delicately controlled, man-made environment, such as an indoor laboratory or a turn-table setup. In order to extend silhouette-based approaches in uncontrolled, natural environments, researchers have explored different possibility to improve the robustness, such as adaptively updating the background model [37,8,23], using a discrete optimization framework [36], proposing silhouette priors over multi-view sets [14], and introducing a sensor fusion scheme to compute an existence probability of the shape [12]. All these proposals address the aforementioned problems in an uncontrolled reconstruction environment.

However, for Shape from Silhouette methods to work in a general environment, there is one more challenge – the occlusions, which can be categorized into two types: (1) Occlusions can happen when a static object blocks the view of a dynamic object, such as the sculpture blocking the person in Fig. 1 (b) & (d). We call the static object an "occluder". Occluders cannot always be removed from the scene in advance, like the sculpture in our example Fig. 1 (a), so their appearances exist as part of the pre-learnt background model. When a dynamic object goes behind a static occluder, since the appearance in the viewing angle does not differ from the background model in this occluded region, an incomplete silhouette happens. Consequently, due to the intersection rule, such corrupted silhouettes result in an incomplete visual hull. This type of occlusion is specific to background-model-and-silhouette-based reconstruction approaches. (2) Occlusions may occur between two or more dynamic objects of interest, as shown in Fig. 1 (c) & (e). We call this "inter-occlusion"s. With the increase of such occlusions, the discriminatory power of the silhouettes decreases, resulting in the reconstructed shapes much larger in volume than the real objects. In fact, when multiple dynamic objects clutter in the scene, the visibility ambiguity in general increases, no matter if two dynamic objects are occluding each other or they are well-separated. We will discuss this in more detail in §2 (Fig. 2.2). Both types of occlusions decrease the quality of the final reconstruction result, yet they are very common and almost unavoidable in natural environments. If we plan to use Shape from Silhouette methods in uncontrolled real-world scenes, we need to solve both types of occlusion problems.

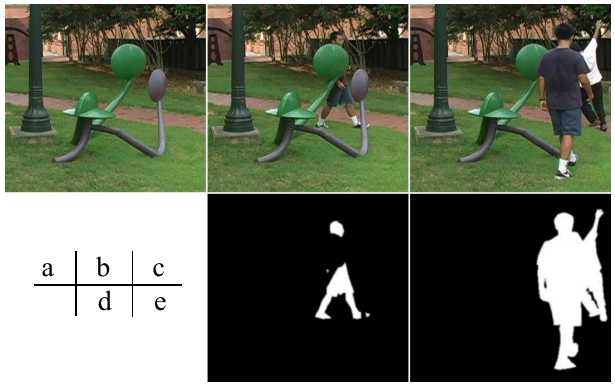

Fig. 1 The occlusion problem for a silhouette-based method. (a) A background view; (b) Occlusion between two dynamic objects; (c) Occlusion between a dynamic object and a static background occluder; (d) Manually segmented silhouette for the two persons' case; (e) Manually segmented silhouette for the person behind the sculpture.

In this paper, we focus on the analysis of visibility relationships. We explicitly model the 3D static occluders in the reconstruction environment. We show that the shape of the static occluders can be recovered incrementally by accumulating occlusion cues from the motion of the dynamic objects. Also by using a distinct appearance model for each dynamic object, inter-occlusion and multi-object visibility ambiguities can be effectively solved. All the reasonings are performed in a Bayesian sensor fusion framework, which is an extension of [12]. Specifically, we use a volume representation of the 3D scene. The major task is to compute the posterior probability for a given voxel to be part of a certain object shape, given multi-view observations. Our algorithm is verified against real datasets to be effective and robust in general outdoor environment of densely populated scenes with possible static occluders.

2 Related Work

2.1 Static Occluder

As shown in the previous section, static occluders make the extracted silhouettes become incomplete, and thus have a negative impact over silhouette-based modeling. In particular the inclusive property of visual hulls [25] with respect to the object being modeled is no longer guaranteed. Generally detecting and accounting for static occlusion has drawn much attention in areas such as depth layer extraction [5], occluding T-junction detection [1], binary occluder mask extraction [15], and single image object boundary interpretation [19]. All these works are limited to 2D image space.

Among papers regarding 3D occlusion, [9] uses sparse 3D occluding T-junctions as salient features to recover structure and motion. In [3], occlusions are implicitly modeled in the context of voxel coloring approaches, using an iterative

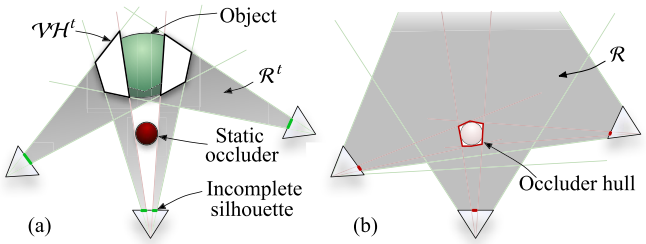

Fig. 2 Deterministic occlusion reasoning. (a) An occluder-free region $\mathcal{R}^t$ can be deduced from the incomplete visual hull $\mathcal{VH}^t$ at time t. (b) $\mathcal{R}$: occluder-free regions accumulated over time.

scheme with semi-transparent voxels and multiple views of a scene from the same time instant. Recently, in the literature of multi-view object tracking [22], a very similar approach to ours is presented, where static occluders are explicitly modeled. The difference is that it uses iterative EM framework that at each frame first solves the voxel occupancy which then feeds back into the system by updating the occlusion model. Hard threshold of silhouette information is required during initialization and the occluder information is maintained in a 4D (a 3D space volume per camera view) state space. Also, the usage of iterative refinement makes it only an offline solution and hard for real-time accelerations.

We represent the static occluder explicitly with a probabilistic 3D volume. What we have observed is as follows: Theoretically occluder shapes can be accessed with careful reasoning about the visual hull of incomplete silhouettes (Fig. 2). Let $\mathcal{S}^t$ be the set of incomplete silhouettes obtained at time t, and $\mathcal{VH}^t$ the incomplete visual hull obtained using these silhouettes. These entities are said to be incomplete because the silhouettes used are potentially corrupt by static occluders that mask the silhouette extraction process. However the incomplete visual hull is a region that is observed by all cameras as being both occupied by an object and unoccluded from any view. Thus we can deduce an entire region $\mathcal{R}^t$ of points in space that are free from any static occluder shape. $\mathcal{R}^t$ is the set of points $X \in \mathbb{R}^3$ for which a view i exists, such that the viewing line of X from view i hits the incomplete visual hull at a first visible point A_i, and $X \in O_iA_i$, with O_i the optical center of view i (Fig. 2(a)). The latter expresses the condition that X appears in front of the visual hull with respect to view i. The region $\mathcal{R}^t$ varies with t, thus assuming static occluders and broad coverage of the scene by dynamic object motion, the free space in the scene can be deduced as the region $\mathcal{R} = \bigcup_{t=1}^{T} \mathcal{R}^t$. The shape of occluders, including concavities if they were covered by object motion, can be recovered as the complement of $\mathcal{R}$ in the common visibility region of all views (Fig. 2(b)).

However this deterministic approach would yield an impractical and non-robust solution, due to inherent silhouette extraction sensitivities to noise and corruption that contribute irreversibly to the result. It also suffers from the limitation that only portions of objects that are seen by all views can contribute to occlusion reasoning. Also, this scheme only accumulates *negative* information, where occluders are certain not to be. However *positive* information is also underlying to the problem: had we known or taken a good guess at where the object shape was (which current shape-from-silhouette methods are able to provide [12]), discrepancies between the object's projection and the actual recorded silhouette would tell us where an occlusion is positively happening. To lift these limitations and provide a robust solution, we propose a probabilistic approach to occlusion reasoning, in which all negative and positive cues are fused and compete in a complementary way toward occluder shape estimation.

2.2 Multiple Dynamic Objects Situation

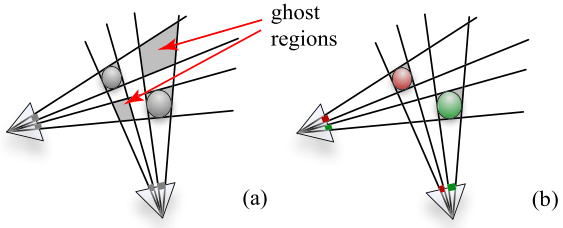

Fig. 3 The principle of multi-object silhouette reasoning for shape modeling disambiguation. (a) Ambiguous "ghost" regions in gray polygons, due to the binary silhouette back-projection does not have enough discriminability. (b) The ghost region ambiguities are eliminated after distinguish between multiple objects' appearances. Best viewed in color.

Most existing silhouette-based reconstruction methods focus on mono-object situations, and fail to address the more general multi-object cases. When multiple dynamic objects are at presence in the scene, besides the inter-occlusion problem in Fig. 1 (c) & (e), binary silhouettes and the consequent visual hull are ambiguous in distinguishing between regions actually occupied by objects and silhouette-consistent "ghost" regions, which occur when regions occupied by objects of interest cannot be disambiguated from free-space regions that also happen to project inside all silhouettes. The polygonal gray region in Fig. 2.2 (a) illustrates this phenomenon. 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.

The "ghost" regions have been analyzed in the context of counting or tracking applications to avoid committing to a "ghost" track [41, 31]. 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 [30, 10]. The inter-occlusion problem has also been studied for the specific case of transparent objects [3]. Recent tracking efforts also use 2D probabilistic inter-occlusion reasoning to improve object localization [18].

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. 2.2(b). It is possible to process multiple silhouette labels in a deterministic, purely geometric fashion [42], 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, 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 [7, 27]. The proposed method can be easily combined with our static occluder recovery. This scheme enables us to perform silhouette inference (§3.2) 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.

In the rest of this paper, we first introduce the fundamental probabilistic sensor fusion framework and the detailed formulations in §3. We then describe extra problems when putting all math expressions together as an automatic system such as appearance automatic initialization and tracking the dynamic objects' motions in §4. Specifically, how to initialize the appearance models and keep track the motion and status of each dynamic object. §5 shows the results of the proposed system and algorithm on completely real-world datasets. Despite the challenges in the datasets, such as lighting variation, shadows, background motion, reflection, dense population, drastic color inconsistency between views, etc, our system produces high quality reconstructions. §6 analyzes the advantages and limitations of this framework and compares the two types of occlusions in more depth, and draws the future picture.

3 Probabilistic Framework

Assume we have a set of calibrated cameras, in this section we introduce our probabilistic shape inference framework in details. With the following notations, we can define our problem formally: given a set of synchronized observations $\mathcal{I}$ from n cameras at a specific time instant, we infer for every discretized location X in an occupancy grid expanding the 3D space its probability of being $\mathcal{L} \in \{\emptyset, 1, \cdots, m, \mathcal{U}, \mathcal{O}\}$. A voxel is either empty ($\emptyset$), one of m objects the model is keeping track of (numerical labels), or occupied by an occluder ($\mathcal{O}$). There is one more label that could be assigned, namely the 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 coming into the scene (§4.3). We denote all the dynamic objects as $\mathcal{G} \in \{1, \cdots, m, \mathcal{U}\}$.

Theoretically, this is a simple posterior probability computation problem, given camera observations. However, in practice, given our huge state space, i.e. the solid 3D volume, and multiple labels, to which every voxel in the volume could be assigned, it is impossible to enumerate all status configurations and find the one with the highest chance given the sensor observations. So before we move on to our detailed formulations, let us take a look at our probabilistic framework, its feasibility to produce the 3D reshape, and the assumptions and simplifications we have, given the specific reconstruction setup.

The joint estimation of the foreground and occluder object shapes would turn the problem into a global optimization over the conjunction of both shape spaces, which becomes intractable, because estimation of a voxel's state bares dependencies with all other voxels on its viewing lines with respect to all cameras. People have encountered similar problems and come up with ideas to deal with this large state space include an EM framework[22], which iteratively converge the state space to an optimal solution, and a solution that decreases the status space into 2D and then solve the global solution [10]. But because we want to recover full 3D information for dynamic scenes, and we would like to keep the possibilities of extending the framework to real-time and online processing, all previous proposals are not satisfactory. Instead, to benefit from the locality that makes occupancy grid approaches practical and efficient, we break the estimation into two steps: for every time instant, we first estimate the occupancy of the dynamic objects from silhouette information using a Bayesian sensor fusion scheme, then es-

timate the occluder occupancy in a second Bayesian inference, using the result of the first estimation as a prior for dynamic objects' occupancy. Although refinements can be achieved by doing iterations over this solution, we demonstrate with real data-sets that the shape estimation is already good enough.

So our main scheme is as follows. For the dynamic objects, the pre-learnt background models and camera sensor models explain which object appearance distribution we are supposed to observe. The models also encode state information about the viewing line through the voxel and potential obstructions from other dynamic objects. For the static occluder, in a separate Bayesian estimation, for each voxel in a 3D grid sampling the acquisition space, we compute how likely it is to be occupied by a static occluder object.

However, for clarity, we are going to describe the static occluder inference first, because it addresses only one problem - the visual occlusions, whereas for the dynamic objects inference, it additionally has many more complicated problems such as appearance learning, object tracking. Also by explaining the static occlusion problem first, it is easier to understand how we treat the inter-occlusion between the dynamic objects. The static occluder is discussed in §3.1 and multiple dynamic objects in §3.2. Following is the main notations that we use in the rest of this paper. Other context-specific notations will be introduced in local sections.

Notations

n	number of cameras
m	number of dynamic objects
X	3D location, in the occupancy grid
$\overline{l}_i$	viewing line of X to view i
$\hat{X}_i$	3D location, on the viewing ray of X, and in front of X with respect to view i
$\check{X}_i$	3D location, on the viewing ray of X, and behind X with respect to view i
$\mathcal{L}$	voxel labels
$\emptyset$	empty space label
$\mathcal{G}$	dynamic object label
$\mathcal{U}$	label for a newcoming dynamic object, whose appearance has not been learnt
$\mathcal{O}$	static occluder label
i	camera index
$\mathcal{I}_i^t$	image from camera i at time t
$\mathcal{B}_i$	camera i's background model
$\mathcal{C}_i^m$	dynamic object m's appearance model in view i
$\mathcal{S}$	silhouette formation hidden variable

3.1 Static Occluder

In this section, to introduce the static occluder formulation, without losing generality, we analyze the case when only one dynamic object in the scene. Later in the result section §5, we show that our static occluder recovery framework also works for multiple dynamic objects cases. So with a single dynamic object in the scene, at voxel X , $\mathcal{G} \in \{1, \cdots, m, \mathcal{U}\}$ can be simplified as one binary label $\mathcal{G}$, namely $\mathcal{G} = 1$ denotes a certain voxel is occupied by the dynamic object, and $\mathcal{G} = 0$ denotes it is not. And the occluder occupancy state at X can also be expressed using the binary label $\mathcal{O}$. $\mathcal{O} = 1$ means the voxel is occupied by the static occluder, $\mathcal{O} = 0$ means it is not. One thing to note is that the occluder state for every voxel is assumed to be fixed over the entire experiment, under the assumption that the occluder is static. Dynamic object status $\mathcal{G}$ on the other hand is not fixed for a voxel, but varies over a number of time instants $t \in \{1, \cdots T\}$ throughout the video frames, where T denotes the last frame acquired so far. In particular, the dynamic object occupancy of voxel X at time t is expressed by a $\mathcal{G}^t$. As shown in Fig. 4(a), the regions of importance to infer both $\mathcal{G}$ and $\mathcal{O}$ are the n viewing lines $\overline{l}_i$, $i \in \{1, \cdots, n\}$ from the camera views to X .

3.1.1 Observed Variables

The voxel X projects to n image pixels $x_i, i \in \{1, \cdots, n\}$, whose color observed at time t in view i is expressed by the variable $\mathcal{I}_i^t$. We assume that background images, which are generally static, were pre-recorded free of dynamic objects, and that the appearance and variability of background colors for pixels x_i was modeled using a set of parameters $\mathcal{B}_i$. Such observations can be used to infer the probability of dynamic object occupancy in the absence of background occluders. The problem of recovering occluder occupancy is more complex because it requires modeling interactions between voxels on the same viewing lines. Relevant statistical variables are shown in Fig. 4(b).

3.1.2 Viewing Line Modeling

Because of potential static occlusions, one must account for other occupancies along the viewing lines of X to infer $\mathcal{O}$. These can be either other static occluder states, or dynamic object occupancies which vary across time. Several such occluders or objects can be present along a viewing line, leading to a number of possible occupancy states for voxels on the viewing line of X . Accounting for the combinatorial number of possibilities for voxel states along X 's viewing line is neither necessary nor meaningful: first because occupancies of neighboring voxels are fundamentally correlated to the presence or the absence of a single common object, second because the main useful information one needs

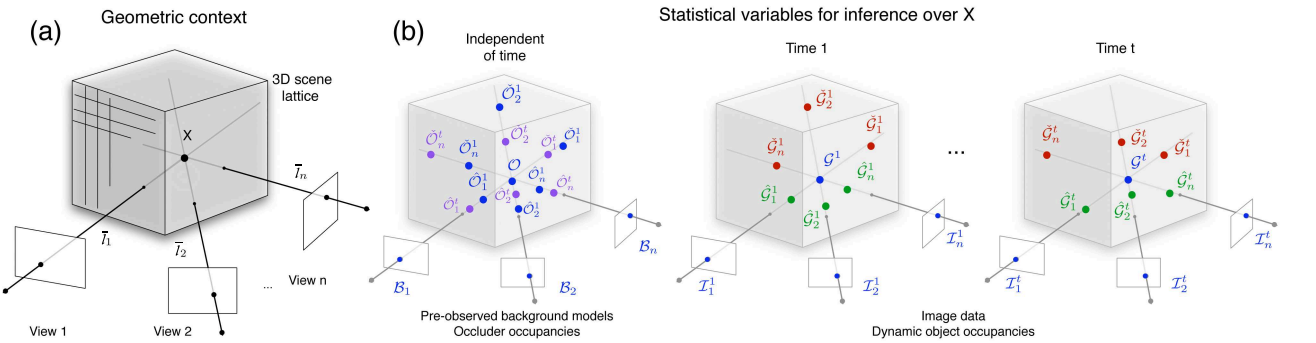

Fig. 4 Problem overview. (a) Geometric context of voxel X. (b) Main statistical variables used to infer the occluder occupancy probability of X. $\mathcal{G}^t$, $\hat{\mathcal{G}}_i^t$, $\check{\mathcal{G}}_i^t$: dynamic object occupancies at relevant voxels at, in front of, behind X respectively. $\mathcal{O}$, $\hat{\mathcal{O}}_i^t$, $\check{\mathcal{O}}_i^t$: static occluder occupancies at, in front of, behind X. $\mathcal{I}_i^t$, $\mathcal{B}_i$: colors and background color models observed where X projects in images.

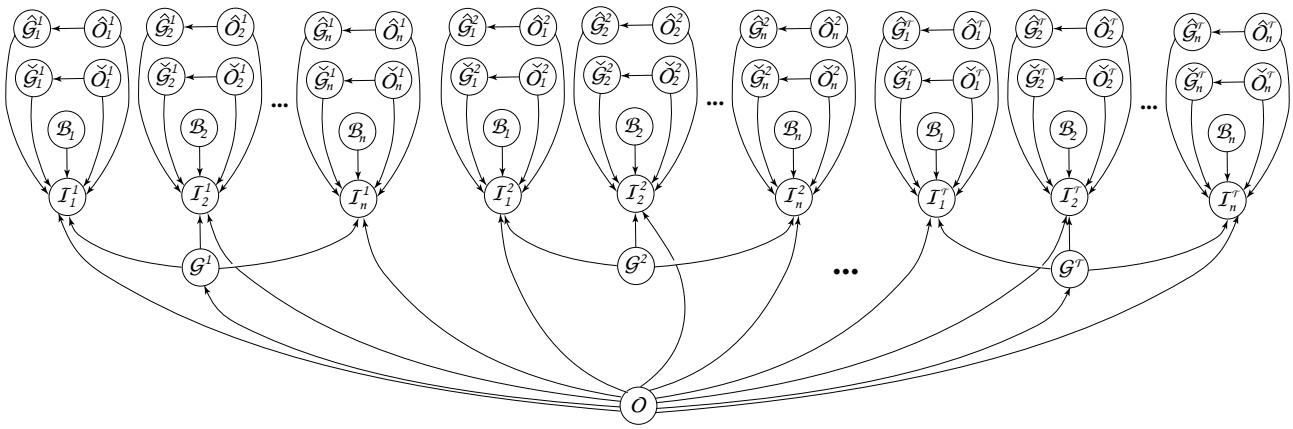

Fig. 5 The dependency graph for the static occluder inference at voxel X , assuming the probability for X to be $\mathcal{G}$ is known. Notice that the background model for each view $\mathcal{B}_i$ does not change with time, but just drawn duplicately for the clarity of the graph.

to know to make occlusion decisions about X is to know whether something is in front of it or behind it, regardless of where along the viewing line.

With this in mind, we model each viewing line using three components, that model the state of X , the state of occlusion of X by anything in front, and the state of what is at the back of X . We model the front and back components by extracting the two most influential modes in front and behind of X , that are given by two voxels $\hat{X}_i^t$ and $\check{X}_i^t$. We select $\hat{X}_i^t$ as the voxel at time t that most contributes to the belief that X is obstructed by a dynamic object along $\overline{l}_i$, and $\check{X}_i^t$ as the voxel most likely to be occupied by a dynamic object behind X on $\overline{l}_i$ at time t.

3.1.3 Viewing Line Unobserved Variables

With this three component modeling, comes a number of related statistical variables illustrated in Fig. 4(b). The occupancy of voxels $\hat{X}_i^t$ and $\check{X}_i^t$ by the visual hull of a dynamic object at time t on $\overline{l}_i$ is expressed by two binary state variables, respectively $\hat{\mathcal{G}}_i^t$ and $\check{\mathcal{G}}_i^t$. Two binary state variables $\hat{\mathcal{O}}_i^t$ and $\check{\mathcal{O}}_i^t$ express the presence or absence of an occluder at voxels $\hat{X}_i^t$ and $\check{X}_i^t$ respectively. Note the difference in semantics between the two variable groups $\hat{\mathcal{G}}_i^t$, $\check{\mathcal{G}}_i^t$ and $\hat{\mathcal{O}}_i^t$, $\check{\mathcal{O}}_i^t$. The former designates dynamic visual hull occupancies of different time instants and chosen positions, while the latter expresses *static* occluder occupancies, whose *position only* was chosen in relation to t. Both need to be considered because they both influence the occupancy inference and are not independent. For legibility, we occasionally refer to the conjunction of a group of variables by dropping indices and exponents, *e.g.* $\mathcal{G} = \{\mathcal{G}_1, \cdots, \mathcal{G}_T\}$, $\mathcal{B} = \{\mathcal{B}_1, \cdots, \mathcal{B}_n\}$.

3.1.4 Joint Distribution

As a further step toward offering a tractable solution to occlusion occupancy inference, we describe the noisy interactions between the variables considered, through the decomposition of their joint distribution $p(\mathcal{O}, \mathcal{G}, \hat{\mathcal{O}}, \hat{\mathcal{G}}, \check{\mathcal{O}}, \check{\mathcal{G}}, \mathcal{I}, \mathcal{B})$. Given the variable dependency graph shown in Fig. 5, we propose the following:

$$p(\mathcal{O})\prod_{t=1}^{T}p(\mathcal{G}^t|\mathcal{O})\prod_{i=1}^{n}p(\hat{\mathcal{O}}_i^t)p(\hat{\mathcal{G}}_i^t\ |\hat{\mathcal{O}}_i^t)p(\check{\mathcal{O}}_i^t)p(\check{\mathcal{G}}_i^t\ |\check{\mathcal{O}}_i^t) \quad (1)$$
$$p(\mathcal{I}_i^t|\hat{\mathcal{O}}_i^t,\hat{\mathcal{G}}_i^t\ ,\mathcal{O},\mathcal{G}^t,\check{\mathcal{O}}_i^t,\check{\mathcal{G}}_i^t\ ,\mathcal{B}_i).$$

$p(\mathcal{O})$, $p(\hat{\mathcal{O}}_i^t)$, and $p(\check{\mathcal{O}}_i^t)$ are priors of occluder occupancy. We set them to a single constant distribution $\mathcal{P}_o$ which reflects the expected ratio between occluder and non-occluder voxels in a scene. No particular region of space is to be favored *a priori*.

3.1.5 Dynamic Occupancy Priors

$p(\mathcal{G}^t|\mathcal{O})$, $p(\hat{\mathcal{G}}_i^t\ |\hat{\mathcal{O}}_i^t)$, $p(\check{\mathcal{G}}_i^t\ |\check{\mathcal{O}}_i^t)$ are priors of dynamic visual hull occupancy with identical semantics. This choice of terms reflects the following modeling decisions. First, the dynamic visual hull occupancies involved are considered independent of one another as they synthesize the information of three distinct regions for each viewing line. However they depend upon the knowledge of occluder occupancy at the corresponding voxel position, because occluder and dynamic object occupancies are mutually exclusive at a given scene location. Importantly however, we do not have direct access to dynamic object occupancies but to the occupancies of its *visual hull*. Fortunately this ambiguity can be adequately modeled in a Bayesian framework, by introducing a local hidden variable $\mathcal{H}$ expressing the correlation between dynamic and occluder occupancy:

$$p(\mathcal{G}^t|\mathcal{O})=\sum_{\mathcal{H}}p(\mathcal{H}\ )p(\mathcal{G}^t|\mathcal{H}\ ,\mathcal{O}). \quad (2)$$

We set $p(\mathcal{H}\ =1)=\mathcal{P}_c$ using a constant expressing our prior belief about the correlation between visual hull and occluder occupancy. The prior $p(\mathcal{G}^t|\mathcal{H}\ ,\mathcal{O})$ explains what we expect to know about $\mathcal{G}^t$ given the state of $\mathcal{H}$ and $\mathcal{O}$:

$$p(\mathcal{G}^t=1|\mathcal{H}\ =0,\mathcal{O}=\omega)=\mathcal{P}_{\mathcal{G}_t}\quad\forall\omega \quad (3)$$
$$p(\mathcal{G}^t=1|\mathcal{H}\ =1,\mathcal{O}=0)=\mathcal{P}_{\mathcal{G}_t} \quad (4)$$
$$p(\mathcal{G}^t=1|\mathcal{H}\ =1,\mathcal{O}=1)=\mathcal{P}_{go}\ , \quad (5)$$

with $\mathcal{P}_{\mathcal{G}_t}$ the prior dynamic object occupancy probability as computed independently of occlusions [12], and $\mathcal{P}_{go}$ set close to 0, expressing that it is unlikely that the voxel is occupied by dynamic object visual hulls when the voxel is known to be occupied by an occluder and both dynamic and occluder occupancy are known to be strongly correlated (5). The probability of visual hull occupancy is given by the previously computed occupancy prior, in case of non-correlation (3), or when the states are correlated but occluder occupancy is known to be empty (4).

3.1.6 Image Sensor Model

The sensor model $p(\mathcal{I}_i^t|\hat{\mathcal{O}}_i^t,\hat{\mathcal{G}}_i^t\ ,\mathcal{O},\mathcal{G}^t,\check{\mathcal{O}}_i^t,\check{\mathcal{G}}_i^t\ ,\mathcal{B}_i)$ is governed by a hidden local per-pixel process $\mathcal{S}$. The binary variable $\mathcal{S}$represents the hidden silhouette detection state (0 or 1) at this pixel. It is unobserved information and can be marginalized, given an adequate split into two subterms:

$$p(\mathcal{I}_i^t|\hat{\mathcal{O}}_i^t,\hat{\mathcal{G}}_i^t\ ,\mathcal{O},\mathcal{G}^t,\check{\mathcal{O}}_i^t,\check{\mathcal{G}}_i^t\ ,\mathcal{B}_i) \quad (6)$$
$$=\sum_{\mathcal{S}}p(\mathcal{I}_i^t|\mathcal{S},\mathcal{B}_i)p(\mathcal{S}|\hat{\mathcal{O}}_i^t,\hat{\mathcal{G}}_i^t\ ,\mathcal{O},\mathcal{G}^t,\check{\mathcal{O}}_i^t,\check{\mathcal{G}}_i^t\ ).$$

$p(\mathcal{I}_i^t|\mathcal{S},\mathcal{B}_i)$ indicates what color distribution we expect to observe given the knowledge of silhouette detection and background color model at this pixel. When $\mathcal{S}=0$, the silhouette is undetected and thus the color distribution is dictated by the pre-observed background model $\mathcal{B}_i$ (considered Gaussian in our experiments). When $\mathcal{S}=1$, a dynamic object's silhouette is detected, in which case our knowledge of color is limited, thus we use a uniform distribution in this case, favoring no dynamic object color *a priori*.

$p(\mathcal{S}|\hat{\mathcal{O}}_i^t,\hat{\mathcal{G}}_i^t\ ,\mathcal{O},\mathcal{G}^t,\check{\mathcal{O}}_i^t,\check{\mathcal{G}}_i^t\ )$ is the second part of the sensor model, which explicits what silhouette state is expected to be observed given the three dominant occupancy state variables of the corresponding viewing line. Since these are encountered in the order of visibility $\hat{X}_i^t$, X , $\check{X}_i^t$, the following relations hold:

$$p(\mathcal{S}|\{\hat{\mathcal{O}}_i^t,\hat{\mathcal{G}}_i^t\ ,\mathcal{O},\mathcal{G}^t,\check{\mathcal{O}}_i^t,\check{\mathcal{G}}_i^t\ \}=\{o,g,\mathrm{k},\mathrm{l},\mathrm{m},\mathrm{n}\},\mathcal{B}_i) \quad (7)$$
$$=p(\mathcal{S}|\{\hat{\mathcal{O}}_i^t,\hat{\mathcal{G}}_i^t\ ,\mathcal{O},\mathcal{G}^t,\check{\mathcal{O}}_i^t,\check{\mathcal{G}}_i^t\ \}=\{0,0,o,g,\mathrm{p},\mathrm{q}\},\mathcal{B}_i)$$
$$=p(\mathcal{S}|\{\hat{\mathcal{O}}_i^t,\hat{\mathcal{G}}_i^t\ ,\mathcal{O},\mathcal{G}^t,\check{\mathcal{O}}_i^t,\check{\mathcal{G}}_i^t\ \}=\{0,0,0,0,o,g\},\mathcal{B}_i)$$
$$=P_S(\mathcal{S}|o,g)\qquad\forall(o,g)\neq(0,0)\ \ \forall(\mathrm{k},\mathrm{l},\mathrm{m},\mathrm{n},\mathrm{p},\mathrm{q}).$$

These expressions convey two characteristics. First, that the form of this distribution is given by the first non-empty occupancy component in the order of visibility, regardless of what is behind this component on the viewing line. Second, that the form of the first non-empty component is given by an identical sensor prior $P_S(\mathcal{S}|o,g)$. We set the four parametric distributions of $P_S(\mathcal{S}|o,g)$ as following:

$$P_S(\mathcal{S}=1|0,0)=\mathcal{P}_{fa}\qquad P_S(\mathcal{S}=1|1,0)=\mathcal{P}_{fa} \quad (8)$$
$$P_S(\mathcal{S}=1|0,1)=\mathcal{P}_{d}\qquad P_S(\mathcal{S}=1|1,1)=0.5, \quad (9)$$

where $\mathcal{P}_{fa}\ \in[0,1]$ and $\mathcal{P}_d\ \in[0,1]$ are constants expressing the prior probability of *false alarm* and the probability of *detection*, respectively. They can be chosen once for all datasets as the method is not sensitive to the exact value of these priors. Meaningful values for $\mathcal{P}_{fa}$ are close to 0, while $\mathcal{P}_d$ is generally close to 1. (8) expresses the cases where

no silhouette is expected to be detected in images, i.e. either when there are no objects at all on the viewing line, or when the first encountered object is a static occluder, respectively. (9) expresses two distinct cases. First, the case where a dynamic object's visual hull is encountered on the viewing line, in which case we expect to detect a silhouette at the matching pixel. Second, the case where both an occluder and dynamic visual hull are present at the first non-free voxel. This is perfectly possible, because the visual hull is an overestimate of the true dynamic object shape. While the true shape of objects and occluders are naturally mutually exclusive, the *visual hull* of dynamic objects can overlap with occluder voxels. In this case we set the distribution to uniform, because the silhouette detection state cannot be predicted: it can be caused by shadows casted by dynamic objects on occluders in the scene, and noise.

3.1.7 Inference

Estimating the occluder occupancy at a voxel translates to estimating $p(\mathcal{O}|\mathcal{IB})$ in Bayesian terms. Applying Bayes rule to the modeled joint probability (1) leads to the following expression, once hidden variable sums are decomposed to factor out terms not required at each level of the sum:

$$p(\mathcal{O}|\mathcal{IB}) = \frac{1}{z}\, p(\mathcal{O}) \prod_{t=1}^{T} \left(\sum_{\mathcal{G}_t} p(\mathcal{G}^t|\mathcal{O}) \left(\prod_{i=1}^{n} \mathcal{P}_i^t \right) \right) \quad (10)$$

$$\text{where } \mathcal{P}_i^t = \sum_{\check{\mathcal{O}}_i^t, \check{\mathcal{G}}_i^t} p(\check{\mathcal{O}}_i^t) p(\check{\mathcal{G}}_i^t \,|\check{\mathcal{O}}_i^t) \sum_{\hat{\mathcal{O}}_i^t, \hat{\mathcal{G}}_i^t} p(\hat{\mathcal{O}}_i^t) p(\hat{\mathcal{G}}_i^t \,|\hat{\mathcal{O}}_i^t)$$
$$p(\mathcal{I}_i^t|\hat{\mathcal{O}}_i^t, \hat{\mathcal{G}}_i^t\,, \mathcal{O}, \mathcal{G}^t, \check{\mathcal{O}}_i^t, \check{\mathcal{G}}_i^t\,, \mathcal{B}_i). \quad (11)$$

$\mathcal{P}_i^t$ expresses the contribution of view i at a time t. The formulation therefore expresses Bayesian fusion over the various observed time instants and available views, with marginalization over unknown viewing line states (10). The normalization constant z is easily obtained by ensuring summation to 1 of the distribution.

3.1.8 Online Incremental Computation

To determine the reliability of voxels, we model the intuition that voxels whose occlusion cues arise from an abnormally low number of views should not be trusted. Since this clause involves all cameras and their observations jointly, the inclusion of this constraint in our initial model would break the symmetry in the inference formulated in (10) and defeat the possibility for online updates. Instead, we opt to use a second criterion in the form of a reliability measure $R \in [0, 1]$. Small values indicate poor coverage of dynamic objects, while large values indicate sufficient cue accumulation. We define reliability using the following expression:

$$R = \frac{1}{n} \sum_{i=1}^{n} \max_t (1 - \mathcal{P}_{\hat{\mathcal{G}}_i^t}) \mathcal{P}_{\check{\mathcal{G}}_i^t} \quad (12)$$

with $\mathcal{P}_{\hat{\mathcal{G}}_i^t}$ and $\mathcal{P}_{\check{\mathcal{G}}_i^t}$ the prior probabilities of dynamic visual hull occupancy. R examines, for each camera i, the maximum occurrence across the examined time sequence of X to be both unobstructed and in front of a dynamic object. This determines how well a given view i was able to contribute to the estimation across the sequence. R then averages these values across views, to measure the overall quality of observation, and underlying coverage of dynamic object motion for the purpose of occlusion inference.

The reliability R can be used online in conjunction to the occlusion probability estimation to evaluate a conservative occluder shape at all times, by only considering voxels for which R exceeds a certain quality threshold. As shown in §5.1.1, it can be used to reduce the sensitivity to noise in regions of space that have only been observed marginally.

3.1.9 Accounting for Occlusion in SfS

As more data becomes available and reliable, the results of occluder estimation can be accounted for when inferring the occupancies of dynamic objects. This translates to the evaluation of $p(\mathcal{G}^\tau|\mathcal{I}^\tau\mathcal{B})$ for a given voxel X and time τ. The difference with the classical single-frame formulation of dynamic object occupancy [12] is that we now have a prior over the occlusions at every voxel in the grid. For this inference $\mathcal{G}^\tau$ is considered independent of $\mathcal{G}^t\ \forall t \neq \tau$, leading to the following simplified joint probability distribution:

$$p(\mathcal{O})p(\mathcal{G}^\tau|\mathcal{O}) \prod_{i=1}^{n} p(\hat{\mathcal{O}}_i^\tau) p(\hat{\mathcal{G}}_i^\tau \,|\hat{\mathcal{O}}_i^\tau) p(\mathcal{I}_i^\tau|\hat{\mathcal{O}}_i^\tau, \hat{\mathcal{G}}_i^\tau\,, \mathcal{O}, \mathcal{G}^\tau, \mathcal{B}_i),$$

where $\mathcal{G}^\tau$ and $\mathcal{O}$ are the dynamic and occluder occupancy at the inferred voxel, $\hat{\mathcal{O}}_i^\tau$, $\hat{\mathcal{G}}_i^\tau$ the variables matching the most influential component along $\overline{l}_i$, *in front of* X . This component is selected as the voxel whose prior of being occupied is maximal, as computed to date by occlusion inference. In this inference, there is no need to consider voxels behind X , because knowledge about their occlusion occupancy has no influence on the state of X .

The parametric forms of this distribution have identical semantics as §3.1.4 but different assignments because of the nature of the inference. Naturally no prior information about dynamic occupancy is assumed here. $p(\mathcal{O})$ and $p(\hat{\mathcal{O}}_i^\tau)$ are set using the result to date of expression (10) at their respective voxels, as prior. $p(\mathcal{G}^\tau|\mathcal{O})$ and $p(\hat{\mathcal{G}}_i^\tau \,|\hat{\mathcal{O}}_i^\tau)$ are constant: $p(\mathcal{G}^\tau{=}1|\mathcal{O}{=}0){=}0.5$ expresses a uniform prior for dynamic objects when the voxel is known to be occluder free. $p(\mathcal{G}^\tau{=}1|\mathcal{O}{=}1){=}\mathcal{P}_{go}$ expresses a low prior of dynamic visual hull occupancy given the knowledge of occluder occupancy,

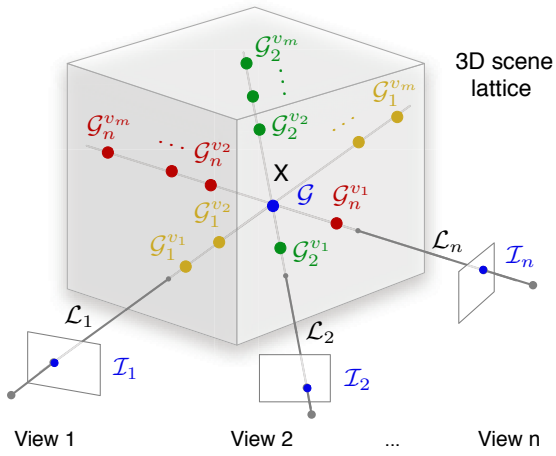

Fig. 6 Overview of main statistical variables and geometry of the problem. $\mathcal{G}$ is the occupancy at voxel X and lives in a state space $\mathcal{L}$of object labels. $\{\mathcal{I}_i\}$ are the color states observed at the n pixels where X projects. $\{\mathcal{G}_i^{v_j}\}$ are the states in $\mathcal{L}$of the most likely obstructing voxels on the viewing line, for each of the m objects, enumerated in their order of visibility $\{v_j\}_i$.

as in (5). The term $p(\mathcal{I}_i^\tau|\hat{\mathcal{O}}_i^\tau, \hat{\mathcal{G}}_i^\tau\ , \mathcal{O}, \mathcal{G}^\tau, \mathcal{B}_i)$ is set same as expression (7), only stripped of the influence of $\check{\mathcal{O}}_i^\tau$, $\check{\mathcal{G}}_i^\tau$.

3.2 Multiple Dynamic Objects

In this section, we focus on the inference of multiple dynamic objects. Since a dynamic object changes shape and location constantly, our dynamic object reconstruction has to be computed for every frame in time, and there is no way to accumulate the information over time as we did for the static occluder. So let's just focus at a single time instant for this section. Our notations slightly change as follows to best describe the formulations: 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 $\overline{l}_i$, $i \in \{1, \cdots, n\}$, with respect to the different views. In this paragraph we assume that some 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 §4. 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. 6.

3.2.1 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 (§4.3).

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 (§3.2.2).

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. 6, 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.

3.2.2 Dependencies Considered

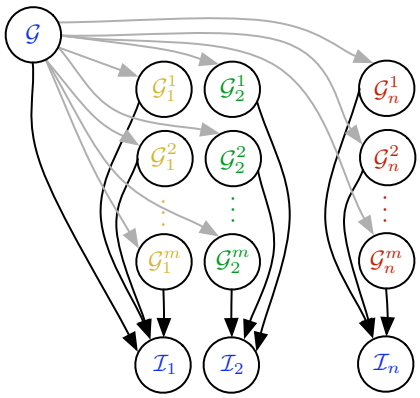

Fig. 7 The dependency graph for the dynamic object inference at voxel X , assuming the probability for X to be other labels are known. Notice that the background model for each view $\mathcal{B}, \mathcal{C}$ and $\mathcal{O}$ are not drawn for simplicity.

Based on the dependency graph Fig. 3.2.2, 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 $p(\mathcal{G}\ \ \mathcal{G}_{1:n}^{1:m}\ \mathcal{I}_{1:n}\ \mathcal{C}_{1:n}^{1:m})$:

$$p(\mathcal{G}\ )\prod_{l\in\mathcal{L}} p(\mathcal{C}_{1:n}^l)\prod_{i,l\in\mathcal{L}} p(\mathcal{G}_i^l|\mathcal{G}\ )\prod_i p(\mathcal{I}_i|\mathcal{G}\ \ \mathcal{G}_i^{1:m}\ \mathcal{C}_i^{1:m}) \tag{13}$$

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 (§4).

$p(\mathcal{C}_{1:n}^l)$ 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 $\mathcal{G} = l$:

$$p(\mathcal{G}_i^k|\{\mathcal{G} = l\}) = \mathcal{P}(\mathcal{G}_i^k) \qquad \text{when } k \neq l \tag{14}$$

$$p(\mathcal{G}_i^l|\{\mathcal{G} = l\}) = \delta_\emptyset(\mathcal{G}_i^l) \qquad \forall l \in \mathcal{L}, \tag{15}$$

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 (15) $p(\mathcal{G}_i^l|\{\mathcal{G} = l\})$ 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. $p(\mathcal{I}_i|\mathcal{G}\ \ \mathcal{G}_i^{1:m}\ \mathcal{C}_i^{1:m})$ is the image formation term. It 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:

$$p(\mathcal{I}_i|\mathcal{G}\ \ \mathcal{G}_i^{1:m}\ \mathcal{C}_i^{1:m}) = \sum_{\mathcal{S}} p(\mathcal{I}_i|\mathcal{S}\ \mathcal{C}_i^{1:m})p(\mathcal{S}|\mathcal{G}\ \ \mathcal{G}_i^{1:m}) \tag{16}$$

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) [37] 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 viewing line $\overline{l}_i$. If $\{g\}_i$ denotes the particular index after which the voxel X itself appears on viewing line $\overline{l}_i$, 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:

$$p(\mathcal{S}|\emptyset\cdots\emptyset\ l\ *\cdots*) = d_l(\mathcal{S}) \quad \text{with } l \neq \emptyset \tag{17}$$

$$p(\mathcal{S}|\emptyset\ \cdots\cdots\ \emptyset) = d_\emptyset(\mathcal{S}), \tag{18}$$

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. (17) 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 ("*"). (18) 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.

3.2.3 Dynamic Object 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 $\mathcal{G}_{1:n}^{1:m}$:

$$p(\mathcal{G}\ |\mathcal{I}_{1:n}\ \mathcal{C}_{1:n}^{1:m}) = \frac{1}{z}\sum_{\mathcal{G}_{1:n}^{1:m}} p(\mathcal{G}\ \ \mathcal{G}_{1:n}^{1:m}\ \mathcal{I}_{1:n}\ \mathcal{C}_{1:n}^{1:m}) \tag{19}$$

$$= \frac{1}{z}p(\mathcal{G}\ )\prod_{i=1}^{n} f_i^1 \tag{20}$$

$$\text{where } f_i^k = \sum_{\mathcal{G}_i^{v_k}} p(\mathcal{G}_i^{v_k}|\mathcal{G}\ ) f_i^{k+1} \quad \text{for } k < m \tag{21}$$

$$\text{and } f_i^m = \sum_{\mathcal{G}_i^{v_m}} p(\mathcal{G}_i^{v_m}|\mathcal{G}\ ) p(\mathcal{I}_i|\mathcal{G}\ \ \mathcal{G}_i^{1:m}\ \mathcal{C}_i^{1:m}) \tag{22}$$

The normalization constant z is easily obtained by ensuring the distribution: $z = \sum_{\mathcal{G}\ ,\mathcal{G}_{1:n}^{1:m}} p(\mathcal{G}\ \ \mathcal{G}_{1:n}^{1:m}\ \mathcal{I}_{1:n}\ \mathcal{C}_{1:n}^{1:m})$. (19) sum up to 1, which 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 (20). 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 (17), 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 (21), $\forall k \in \{1,\cdots,m-1\}$:

$$f_i^k = p(\mathcal{G}_i^{v_k}=\emptyset|\mathcal{G}\ ) f_i^{k+1} + \sum_{l\neq\emptyset} p(\mathcal{G}_i^{v_k}=l|\mathcal{G}\ )\mathcal{P}_l(\mathcal{I}_i) \tag{23}$$

4 Automatic Learning and Tracking

We have presented in §3 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 §4.1. Intuitively, object shape estimation and tracking are complementary and mutually helpful tasks. We explain in §4.2 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 (§4.3). Finally, static occluder computation can easily be integrated in the system and help the inference be robust to static occluders (§4.4). The algorithm at every time instance is summarized in Alg. 1.

Algorithm 1 Dynamic Scene Reconstruction

Input: Frames at a new time instant for all views
Output: 3D object shapes in the scene
Coarse Inference;
if a new object enters the scene **then**
  add a label for the new object;
  initialize foreground appearance model;
  go back to **Coarse Inference**;
end if
Refined Inference;
static occluder inference;
update object location and prior;
return

4.1 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

$$p(\mathcal{G}\ |\mathcal{I}_{1:n}\ \mathcal{C}_{1:n}^{1:m}) = \frac{1}{z}p(\mathcal{G}\ )\prod_{i=1}^{n} p(\mathcal{I}_i|\mathcal{G}\ \ \mathcal{C}_i^{1:m}) \tag{24}$$

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 (14) 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 §4.2.

4.2 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 (§5.2.2), thanks to the rich shape and occlusion information modeled.

4.3 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 §4.2. 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
- $\mathcal{U}$ is not too close to the scene boundary
- Subsequent updates of $\mathcal{U}$'s track are bounded

The first criterion is very straightforward. The second one guarantees that the object is not any kind of consistent noise over all views, or any moving objects that is too small to be of our interest. The third one guarantees that the object we are trying to model is likely to have full observations from all of the views. The fourth item further eliminates random noises. One thing to note is that, even with all the above criterae, it does sometimes happen that the object we are trying to model is currently occluded by either another dynamic object or a static occluder to some of the views. But luckily, our appearance initialization is view-based, and thus does not require all views aquire the object appearance at the same time. This is another advantage of using view-based appearance than a global appearance, besides the previously discussed fact that we can bypass the tedius radiometric calibration of the network. However, only when all the views have finished the appearance initialization of a certain object, a new label is added to $\mathcal{L}$.

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 §5.2.1.

4.4 Occluder computation

The static occluder computation can easily be integrated with the multiple dynamic object reconstruction described in §3.1. 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 §3.1. It could be thought that the multi-label dynamic object inference discussed in this section is an extension to the single dynamic object cases assumed in §3.1. In fact, the occlusion occupancy inference does benefit from the disambiguation inherent to multi-silhouette reasoning, as the real-world experiment shows, in Fig. 16, in §5.

5 Result and Evaluation

5.1 Occlusion Inference Results

To demonstrate the power of the static occluder shape recovery, we mainly use a single person as the dynamic object in the scene. In the next section, we also show that it can be recovered in the presence of multiple dynamic objects. We show three multi-view sequences: the PILLARS and SCULPTURE sequences, acquired outdoors, and the CHAIR sequence, acquired indoors, with combined artificial and natural light from large bay windows. In all sequences nine DV cameras surround the scene of interest, background models are learned in the absence of moving objects. A single person as our dynamic object walks around and through the occluder in each scene. The shape of the person is estimated at each considered time step and used as prior to occlusion inference. The data is used to compute an estimate of the occluder's shape using (10). Results are presented in Fig. 8.

Nine geometrically calibrated 720×480 resolution cameras all record at 30Hz. Color calibration is unnecessary because the model uses silhouette information only. The background model is learned per-view using a single Gaussian color model per pixel, and training images. Although simple, the model proves sufficient, even in outdoor sequences subject to background motion, foreground object shadows,

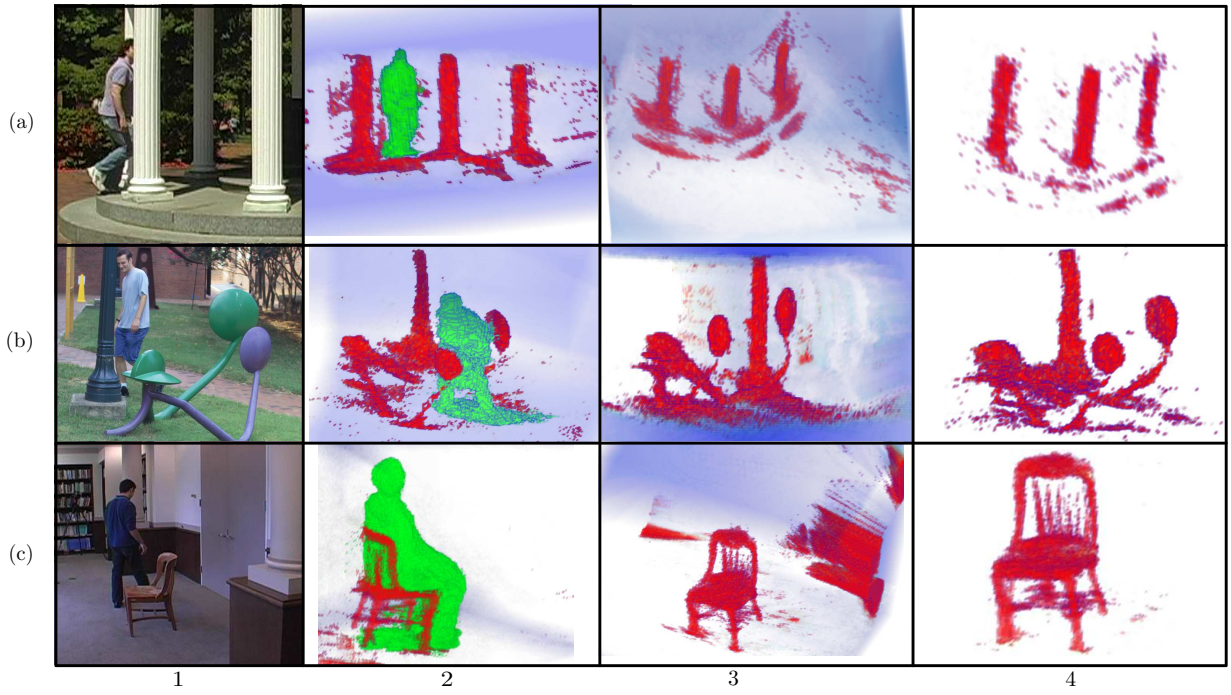

Fig. 8 Occluder shape retrieval results. Sequences: (a) PILLARS , (b) SCULPTURE , (c) CHAIR . 1) Scene overview. Note the harsh light, difficult backgrounds for (a) and (b), and specularity of the sculpture, causing no significant modeling failure. 2-3) Occluder inference according to Blue: neutral regions (prior $\mathcal{P}_o$), red: high probability regions. Brighter/clear regions indicate the inferred absence of occluders. Fine levels of detail are modeled, sometimes lost - mostly to calibration. In (a) the structure's steps are also detected. 4) Same inference with additional exclusion of zones with reliability under 0.8. Peripheral noise and marginally observed regions are eliminated. The background protruding shape in (c3) is due to a single occlusion from view (c1). The supplemental video shows extensive results with these datasets, including one or more people in the scene.

and substantial illumination changes, illustrating the strong robustness of the method to difficult real conditions. The method can cope well with background misclassifications that do not lead to large coherent false positive dynamic object estimations: pedestrians are routinely seen in the background for the SCULPTURE and PILLARS sequences (e.g. Fig. 8(a1)), without any significant corruption of the inference.

Adjacent frames in the input videos contain largely redundant information for occluder modeling, thus videos can safely be subsampled. PILLARS was processed using 50% of the frames (1053 frames processed), SCULPTURE and CHAIR with 10% (160 and 168 processed frames respectively).

5.1.1 Online Computation Results

All experiments can be computed using incremental inference updates. Fig. 9 depicts the inference's progression, using the sensor fusion formulation alone or in combination with the reliability criterion. For the purpose of this experiment, we used the PILLARS sequence and manually segmented the occluder in each view for a ground truth comparison, and focused on a subregion of the scene in which the expected behaviors are well isolated. Fig. 9 shows that both schemes converge reasonably close to the visual hull of the considered pillar. In scenes with concave parts accessible to dynamic objects, the estimation would carve into concavities and reach a better estimate than the occluder's visual hull. A somewhat larger volume is reached with both schemes in this example. This is attributable to calibration errors which overtightens the visual hull with respect to the true silhouettes, and accumulation of errors in both schemes toward the end of the sequence. We trace those to the redundant, periodical poses contained in the video, that sustain consistent noise. This suggests the existence of an optimal finite number of frames to be used for processing. Jolts can be observed in both volumes corresponding to instants where the person walks behind the pillar, thereby adding positive contributions to the inference. Use of the reliability criterion contributes to lower sensitivity to noise, as well as a permanently conservative estimate of the occluder volume as the curves show in frames 100-200. Raw inference (10) momentarily yields large hypothetical occluder volumes when data is biased toward contributions of an abnormally low subset of views (frame 109).

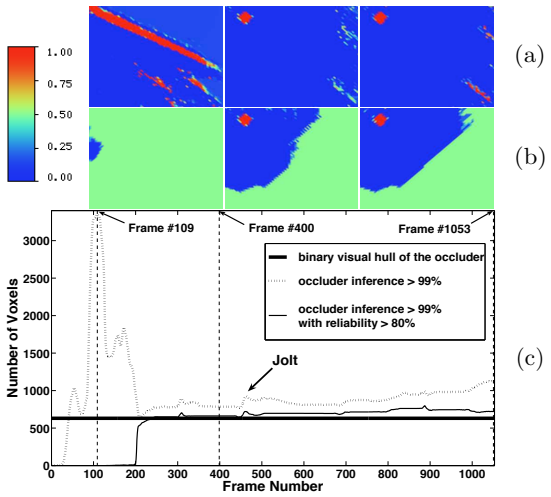

Fig. 9 Online inference analysis and ground truth visual hull comparison, using PILLARS dataset, focusing on a slice including the middle pillar (**best viewed in color**). (a) Frames 109, 400 and 1053, inferred using (10). (b) Same frames, this time excluding zones with reliability under 0.8 (reverted here to 0.5). (c) Number of voxels compared to ground truth visual hull across time.

5.1.2 Accounting for Occlusion in SfS

Our formulation (§3.1.9) can be used to account for the accumulated occluder information in dynamic shape inference. We only use occlusion cues from reliable voxels ($R > 0.8$) to minimize false positive occluder estimates, whose excessive presence would lead to sustained errors. While in many cases the original dynamic object formulation [12] performs robustly, a number of situations benefit from the additional occlusion knowledge (Fig. 10). Person volume estimates can be obtained when accounting for occluders. These estimates appear on average to be a stable multiple of the real volume of the person, which depends mainly on camera configuration. This suggests a possible biometrics application of the method, for disambiguation of person recognition based on computed volumes.

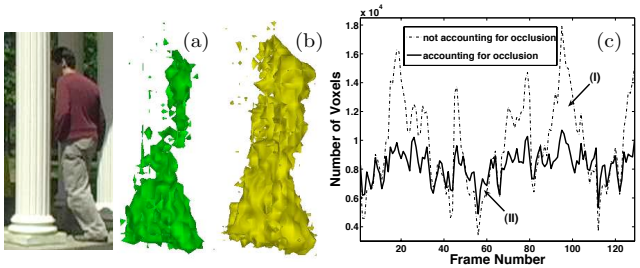

Fig. 10 (a) Person shape estimate from PILLARS sequence, as occluded by the rightmost pillar and computed without accounting for occlusion. (b) Same situation accounting for occlusion, showing better completeness of the estimate. (c) Volume plot in both cases. Accounting for occlusion leads to more stable estimates across time, decreases false positives and overestimates due to shadows cast on occluders (I), increases estimation probabilities in case of occlusion (II).

5.2 Multi-Object Shape Inference Results

We have used four multi-view sequences to validate multi-object shape inference. Eight 30Hz 720×480 DV cameras surrounding the scene in a semi-circle were used for the CLUSTER and BENCH sequences. The LAB sequence is provided by [18] and SCULPTURE was used to reconstruct the static sculpture (Fig. 8(b)) in the previous section. Here, we show the result of multiple persons walking in the scene together with the reconstructed sculpture.

	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.

5.2.1 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. 11, 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 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. But this online updating of the appearance models could be an easy extension for robustness.

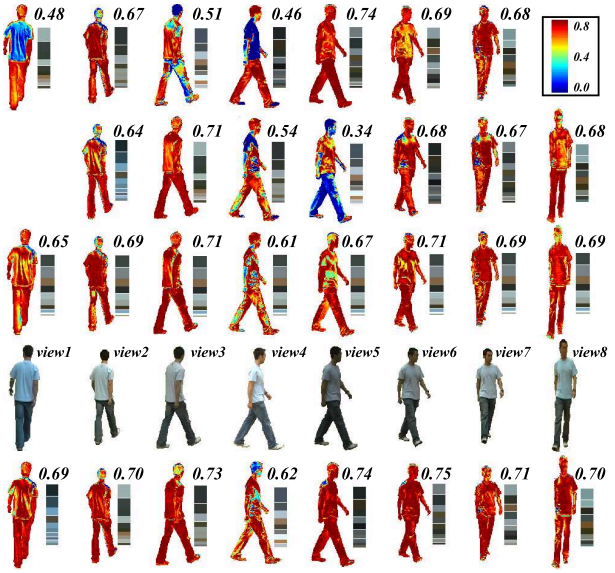

Fig. 11 Appearance model analysis. A person in eight views is displayed in row 4. A GMM model $\mathcal{C}_i$ is trained for view $i \in [1, 8]$. A global GMM model $\mathcal{C}_0$ over all views is also trained. Row 1, 2, 3 and 5 compute $\mathcal{P}(\mathcal{S}|\mathcal{I}, \mathcal{B}, \mathcal{C}_{i+1})$, $\mathcal{P}(\mathcal{S}|\mathcal{I}, \mathcal{B}, \mathcal{C}_{i-1})$, $\mathcal{P}(\mathcal{S}|\mathcal{I}, \mathcal{B}, \mathcal{C}_0)$ 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.

One more thing to note, is that in our approach, even though an object's appearances are learnt for eavh view separately, they are still linked together in 3D by the same object label. In this sense, our per-view based appearances can be taken as an intermediate model between the global model as used in Shape-from-Photoconsistency and multi-view stereo, and the pure 2D image models used by video surveillance and tracking literatures.

5.2.2 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. 12. The naive 2-label reconstruction (probabilistic visual hull) yields large volumes with little separation between objects, because the entire scene configuration 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.

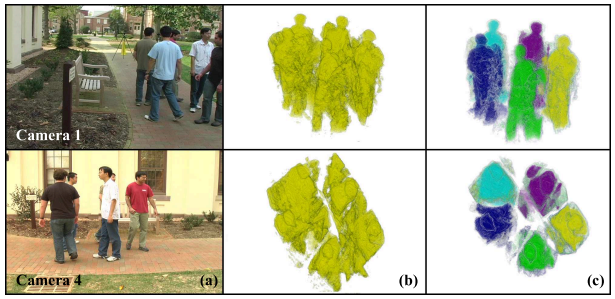

Fig. 12 Result from 8-view CLUSTER dataset. (a) Two views at frame 0. (b) Respective 2-labeled reconstruction. (c) More accurate shape estimation using our algorithm.

The LAB sequence [18] with poor image contrast is also processed. The reconstruction result from all 15 cameras is shown in Fig. 13. Moreover, in order to evaluate our localization prior estimation, we compare our tracking method (§4.2) with the ground truth data, the result of [18] and [30]. We use the exactly same eight cameras as in [30] for the comparison, shown in Fig. 13(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.

5.2.3 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. 14 and the reconstructions at two time instants are shown in Fig. 15.

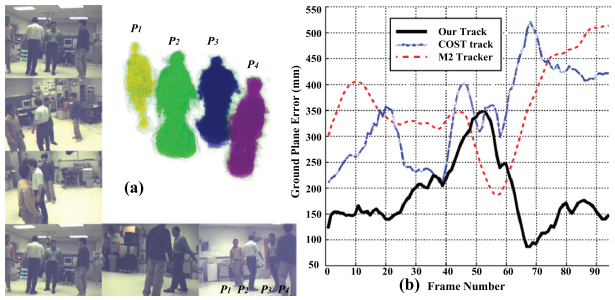

Fig. 13 LAB dataset result from [18]. (a) 3D reconstruction with 15 views at frame 199 (b) 8-view tracking result comparison with methods in [18], [30] and the ground truth data. Mean error in ground plane estimate in mm is plotted.

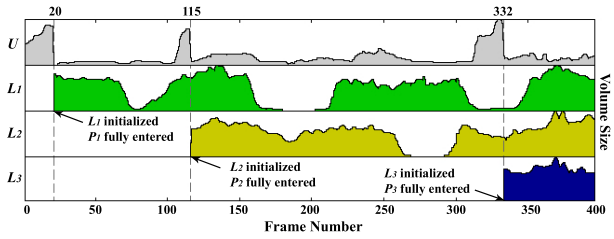

Fig. 14 Appearance model automatic initialization with the BENCH sequence. The volume of $\mathcal{U}$ increases if a new person enters the scene. When an appearance model is learned, a new label is initialized. During the sequence, L_1 and L_2 volumes drop to near zero because they walk out of the scene on those occasions.

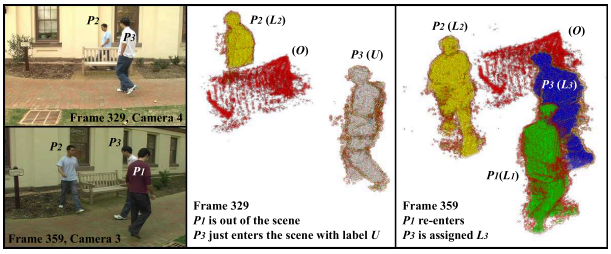

Fig. 15 BENCH result. Person numbers are assigned according to the order their appearance models are initialized. At frame 329, P_3 is entering the scene. Since it's P_3's first time into the scene, he is captured by label $\mathcal{U}$ (gray color). P_1 is out of the scene at the moment. At frame 359, 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.

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.

5.2.4 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. 15 between frame 329 and 359. Check Fig. 14 about the scene configuration during that period. The complete sequence is also given in the supplemental video.

We also compute result for SCULPTURE sequence with two persons walking in the scene, as shown in Fig. 16. For the dynamic objects, we manage to get much cleaner shapes when the two persons are close to each other, and more detailed shapes such as extended arms. For the occluder, thanks to the multiple foreground modes and the consideration of inter-occlusion between the dynamic objects in the scene, we are able to recover the fine shape too. Otherwise, the occluder inference would have to use ambiguous regions when people are clustered.

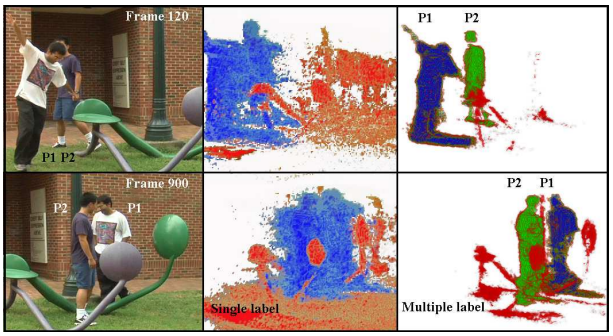

Fig. 16 SCULPTURE data set comparison. The middle column shows the reconstruction with a single foreground label. The right column shows the reconstruction with a label for each person. This figure shows, by resolving inter-occlusion ambiguities, both the static occluder and dynamic objects achieve better quality.

6 Discussion

6.1 Dynamic Object & Static Occluder Comparison

So far, we have shown the mathematical models and real-datasets for static and dynamic objects inference. Although both types of entities are computed only from silhouette cues from camera views and both require the consideration of visual occlusion effect, they actually have fundamentally different characteristics.

First of all, there is no way to learn an appearance model for a static occluder, because it's appearance is initially embedded in the background model of a certain view. Only when an occlusion event happens between the dynamic object and the occluder, can we detect that certain appearance should belong to the occluder but not the background, and the occluder probability should increase along that viewing direction. Whereas for dynamic objects, we have mentioned and will show in more detail in the next section, that their appearance models for all camera views could be manually or automatically learnt before reconstruction.

Secondly, for an occluder, because it is static, places in the 3D scene that has been recovered as highly probable to be occluder will always maintain the high probabilities, not considering noise. This enables the accumulation of the static occluder in our algorithm. But for the inter-occlusion between dynamic objects, it is just a one time instant event. This effect is actually reflected in the inference formulae of the static occluder and the dynamic objects.

Thirdly, a recovered dynamic object can be think of as a probabilistic visual hull representation, because it is after all

a fusion of silhouette information, based on [12]. However, the static occluder that we recover is actually not a visual hull representation. In fact, it is closer to an entity that is carved out using moving visual hulls (of the dynamic objects), as shown in Fig. 2. Therefore, our estimated occluder shape can maintain some concavities, as long as a dynamic object that we use to infer the occluder can move into the concave regions and be witnessed by camera views.

Fig. 17 The static occluder recovered from the SCULPTURE dataset is textured using the observations from the camera views. Two different views are shown here. Best viewed in color.

6.2 Computation Complexity and Acceleration

The occluder occupancy computation was handled on a 2.8 GHz PC at approximately 1 timestep per minute. The very strong locality inherent to the algorithm and preliminary benchmarks suggest that real-time performance could be achieved using a GPU implementation. Occluder information does not need to be processed for every frame because of adjacent frame redundancy, opening the possibility for online, asynchronous cooperative computation of occluder and dynamic objects at interactive frame rates.

The time complexity of our complete system is bounded by the dynamic object inference, where viewing ray maximum probabilities for each label need and each view need to be know. This means a computation of $\mathcal{O}(nmV)$, with n the number of cameras, m the number of objects in the scene, and V the scene volume resolution. We process the dynamic object sequences on a 2.4 GHz Core Quad PC with computation times varying of 1-4 min per time step. Again, the very strong locality inherent to the algorithm and preliminary benchmarks suggest that around 10 times faster performance could be accomplished by a GPU acceleration.

6.3 Shape Refinement

After we get the probabilistic volume of the shapes, we can define surface smoothness and minimum curvature constrains, and use existing global optimization schemes [36, 40] to extract a surface representation of the shape. We can also put textures from the observations to the reconstructed shapes for better visualization and further applications. The sculpture from one of our datasets is textured in Fig. 17.

6.4 Drawbacks and Limitations

There are a few limitations to our approach. First of all, although the static occluder estimation is robust in a general outdoor environment, it is not generally an alternative for static object reconstruction purpose (although it works in some cases, like the CHAIR sequence). This is because our occluder inference is only based on occlusion cues, meaning if there is no occlusion between a dynamic object with the static occluder in a view, we cannot discover the occluder shape. This is why we cannot recover the top of the pillar and lamp post in Fig. 8. However, for dynamic scene analysis, our main focus is on the dynamic objects, in this case, our recovered knowledge about where a dynamic object may possibly be occluded by a static occluder is very important.

Secondly, when dealing with visibilities, either static occluder inference or dynamic objects computation, we have an implicit assumption that the occlusion is partial, namely we still have high confidence what label a certain voxel should be assigned given majority of observation agreement among the non-occluded views. This means we cannot recover a person hiding in a dense crowd, and we cannot recover a solid wall if no views can see a person going behind it. The extreme cases are still remaining to further analysis. In other words, it is also a good question to ask, in order to use our proposed method in a certain scenario, how many cameras would be enough, and how to place them in the scene. But these questions belong to a totally different topic and is beyond the scope of our discussion here.

Finally, the appearance models can be improved. But if two persons with similar color appearances are in the scene, this is a fundamental problem to our scheme, no matter what kind of appearances we use. It will always introduce ambiguities to our dynamic object inference scheme. In this case, the proposed tracking scheme and object location prior will be the main solution. However, the tracking scheme used in the multiple dynamic object inference section is naive. The cylindrical object location prior is not general enough, especially our dynamic object as humans can deform. These are possibilities for future extensions, besides decent ways to further use temporal consistency cues for better dynamic shape estimation.

7 Summary

In this paper, we have presented a complete approach to reconstruct 3D shapes in a dynamic event from silhouettes extracted from multiple videos recorded using a geometrically calibrated camera network. The key elements of our

approach is a probabilistic volumetric framework for automatic 3D dynamic scene reconstruction. The proposed method is robust to occlusion, lighting variation, shadows etc. It does not require photometric calibration among the cameras in the system. It automatically learns the appearance of the dynamic objects, tracks the motions and detects survivance events such as entering/leaving the scene. It also automatically discovers the static occluder, whose appearance is initially hidden in the background and recovers its shape by observing the dynamic objects' performance in the scene for a certain amount of time. Combining all the algorithms described in this paper, it is possible to develop a fully automatic and robust system for dynamic scene analysis in general uncontrolled indoor/outdoor environment.

Acknowledgements We would like to thank A. Gupta *et.al.* [18,30] for providing us the 16-camera dataset. We also gratefully acknowledge the support of David and Lucille Packard Foundation Fellowship and NSF Career award IIS-0237533.

References

1. N. Apostoloff and A. Fitzgibbon. Learning spatiotemporal T-junctions for occlusion detection. *CVPR*, II, p. 553–559, 2005.
2. B.G. Baumgart. *Geometric modeling for computer vision*. PhD thesis, 1974.
3. J. S. De Bonet and P. Viola. Roxels: Responsibility weighted 3d volume reconstruction. *ICCV*, vol. I, p. 418–425, 1999.
4. A. Broadhurst, T. Drummond, and R. Cipolla. A probabilistic framework for the Space Carving algorithm. *ICCV*, p. 388–393, 2001.
5. G. Brostow and I. Essa. Motion based decompositing of video. *ICCV*, p. 8–13, 1999.
6. G. Cheung, T. Kanade, J.-Y. Bouguet, and M. Holler. A real time system for robust 3d voxel reconstruction of human motions. *CVPR*, II:714 – 720, 2000.
7. 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.
8. A. Elgammal, R. Duraiswami, D. Harwood, and L. Davis. Background and foreground modeling using nonparametric kernel density estimation for visual surveillance. *PIEEE*, 90:1151 – 1163, 2002.
9. P. Favaro, A. Duci, Y. Ma, and S. Soatto. On exploiting occlusions in multiple-view geometry. *ICCV*, p. 479–486, 2003
10. F. Fleuret, J. Berclaz, R. Lengagne, and P. Fua. Multi-camera people tracking with a probabilistic occupancy map. *PAMI*, 2007.
11. J.-S. Franco and E. Boyer. Exact polyhedral visual hulls. *BMVC*, pages 329–338, September 2003.
12. J.-S. Franco and E. Boyer. Fusion of multi-view silhouette cues using a space occupancy grid. *ICCV*, II:1747–1753, 2005.
13. Y. Furukawa and J. Ponce. Carved visual hulls for image-based modeling. *ECCV*, 2006.
14. K. Grauman, G. Shakhnarovich, and T. Darrell. A bayesian approach to image-based visual hull reconstruction. *CVPR*, vol. I, p. 187–194, 2003.
15. L. Guan, S. Sinha, J.-S. Franco, and M. Pollefeys. Visual Hull Construction in the Presence of Partial Oclcusion. *3DPVT*, 2006.
16. L. Guan, J.-S. Franco, and M. Pollefeys. 3D Occlusion Inference from Silhouette Cues. *CVPR*, 2007.
17. L. Guan, J.-S. Franco, and M. Pollefeys. Multi-Object Shape Estimation and Tracking from Silhouette Cues. *CVPR*, 2008.
18. A. Gupta, A. Mittal, and L. S. Davis. Cost: An approach for camera selection and multi-object inference ordering in dynamic scenes. *ICCV*, 2007.
19. D. Hoiem, A. Stein, A. Efros, and M. Hebert. Recovering Occlusion Boundaries from a Single Image. *ICCV*, 2007.
20. A. Ilie, and G. Welsh. Ensuring color consistency across multiple cameras. *ICCV*, 2005.
21. N. Joshi, B. Wilburn, V. Vaish, M. Levoy, and M. Horowitz. Automatic Color Calibration for Large Camera Arrays. *UCSD CSE Tech Report CS2005-0821*, 2005.
22. M. Keck, and J. Davis. 3D Occlusion Recovery using Few Cameras. *CVPR*, 2008.
23. K. Kim, D. Harwood, and L. Davis. Background Updating for Visual Surveillance. *ISVC*, 337 – 346, 2005.
24. K. Kutulakos, and S. Seitz. A Theory of Shape by Space Carving. *IJCV*, 2000.
25. A. Laurentini. The visual hull concept for silhouette-based image understanding. *PAMI*, 16(2):150–162, February 1994.
26. S. Lazebnik, E. Boyer, and J. Ponce. On computing exact visual hulls of solids bounded by smooth surfaces. *CVPR*, pages I:156–161, 2001.
27. D. Margaritis and S. Thrun. Learning to locate an object in 3d space from a sequence of camera images. *ICML'98*, 1998.
28. W. Matusik, C .Buehler, R .Raskar, S .Gortler, and L .McMillan. Image-based visual hulls. *Siggraph*, 369–374, 2000.
29. W. Matusik, C. Buehler, and L. Mcmillan. Polyhedral visual hulls for real-time rendering. *Proceedings of Eurographics Workshop on Rendering*, pages 115–126, 2001.
30. 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.
31. K. Otsuka and N. Mukawa. Multiview occlusion analysis for tracking densely populated objects based on 2-D visual angles. *CVPR*, I:90–97, 2004.
32. D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IJCV*, 47:7–42, 2002.
33. S. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. *CVPR*, 2006.
34. S. Sinha and M. Pollefeys. Multi-view reconstruction using photo-consistency and exact silhouette constraints: A maximum-flow formulation. *ICCV*, 2005.
35. G. Slabaugh, B.W. Culbertson, T. Malzbender, M.R. Stevens, and R. Schafer. Methods for volumetric reconstruction of visual scenes. *IJCV*, 57:179–199, 2004.
36. D. Snow, P. Viola, and R. Zabih. Exact voxel occupancy with graph cuts. *CVPR*, p. 345–353, 2000.
37. C. Stauffer and W. E. L. Grimson. Adaptive background mixture models for real-time tracking. *CVPR*, II:246–252, 1999.
38. R. Szeliski, D. Tonnesen, and D. Terzopoulos. Modeling Surfaces of Arbitrary Topology with Dynamic Particles. *CVPR*, p. 82–87, 1993.
39. J. Takamatsu, Y. Matsushita, and K. Ikeuchi. Estimating Camera Response Functions using Probabilistic Intensity Similarity. *CVPR*, 2008.
40. R. Whitaker, D. Breen, K. Museth, and N. Soni. A Framework for Level Set Segmentation of Volume Datasets. *Volume Graphics*, 2001.
41. D. Yang, H. Gonzlez-Baos, and L. Guibas. Counting People in Crowds with a Real-Time Network of Simple Image Sensors. *ICCV*, 2003.
42. R. Ziegler, W. Matusik, H. Pfister, and L. McMillan. 3d reconstruction using labeled image regions. *EG symposium on Geometry processing*, p. 248–259, 2003.