Silhouette Segmentation in Multiple Views

Wonwoo Lee, Woontack Woo, Edmond Boyer

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.
IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE

Silhouette Segmentation in Multiple Views

Wonwoo Lee, *Student Member, IEEE,* Woontack Woo, *Member, IEEE,* and Edmond Boyer

***Abstract*— In this paper, we present a method for extracting consistent foreground regions when multiple views of a scene are available. We propose a framework that automatically identifies such regions in images under the assumption that, in each image, background and foreground regions present different color properties. To achieve this task, monocular color information is not sufficient and we exploit the spatial consistency constraint that several image projections of the same space region must satisfy. Combining monocular color consistency constraint with multi-view spatial constraints allows to automatically and simultaneously segment the foreground and background regions in multi-view images. In contrast to standard background subtraction methods, the proposed approach does not require *a priori* knowledge of the background nor user interaction. Experimental results under realistic scenarios demonstrate the effectiveness of the method for multiple camera setups.**

***Index Terms*— Background region, foreground region, multi-view silhouette consistency, silhouette segmentation**

I. Introduction

IDENTIFYING foreground regions in a single or multiple images is a preliminary step required in many computer vision applications, such as object tracking, motion capture, image and video synthesis, and image-based 3D modeling. In particular, several 3D modeling applications rely on initial models obtained using silhouettes extracted as foreground image regions, e.g., [1]–[3]. Traditionally, foreground regions are segmented under the assumption that the background in each image is static and known beforehand and this operation is usually performed on an individual basis, even when multiple images of the same scene are considered. In this paper, we present a method that extracts consistent foreground regions from multi-view images without *a priori* knowledge of the background. The interest arises in several applications where multi-view images are considered and where information on the background is not reliable or not available.

The approach described in this paper relies on two assumptions that are often satisfied: (i) the region of interest appears entirely in all images; (ii) background colors are consistent in each image, i.e., background colors are different from foreground colors and they are also homogeneous over background pixels. Under these assumptions, we iteratively segment each image such that each background region satisfies color consistency constraints and such that all foreground regions correspond to the same space region. To initiate this iterative process, we exploit the first assumption to identify regions in the images that necessarily belong to background. Such regions are simply image regions that are outside the projections of the observation volume common to all considered viewpoints. These initial regions are then grown iteratively by estimating each pixel's occupancy based on its color and spatial consistencies. This operation can be seen as an estimation of foreground and background parameters given image information with latent variables denoting the region a pixel belongs to, background or foreground. For this task, we adopt an iterative scheme where the background and foreground models are updated in one step and the images are segmented in a subsequent step using the new model parameters. Important features of the approach are as follows: (i) our method is fully automatic and does not require *a priori* knowledge of any type nor user interaction; (ii) images can come either from a single camera at different locations or multiple cameras. In the latter case, cameras do not need to be color-calibrated since color consistency is not enforced among different viewpoints. The overall procedure of the proposed silhouette segmentation method is outlined in Fig. 1.

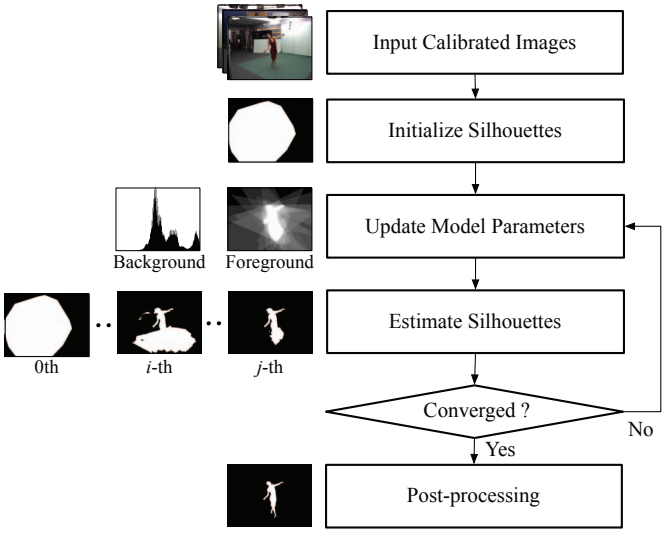


Fig. 1: Approach Outline: first silhouettes are initialized with the projection of the camera visibility domain; Then background and foreground model are iteratively updated and silhouette are re-estimated at each iteration using both color and spatial consistency constraints. Once the optimization is completed, a post-processing step is performed to refine the estimated silhouettes.

The remainder of the paper is as follows. In Section II, we review existing segmentation methods. Section III presents the probabilistic framework within which we model the problem. Section IV details the iterative scheme that is implemented to identify silhouettes. Quantitative and qualitative evaluations are presented in Section V before concluding in Section VI.

II. Related Works

Typical background subtraction methods assume that background pixel values are constant over time, whereas foreground pixel values can vary. Based on this fact, several approaches that take into account photometric information such as grayscale, color, texture or image gradient, have been proposed in a monocular context. *Chroma-keying* approaches belong to this category

Wonwoo Lee and Woontack Woo are with the Gwangju Institute of Science and Technology in S. Korea.
Edmond Boyer is with the LJK - INRIA Rhône Alpes in France

Digital Object Indentifier 10.1109/TPAMI.2010.196

and assume a uniform background, usually blue or green. For non-uniform backgrounds, statistical models are pre-computed for pixels, and the foreground pixels are then identified by comparing current model values. Several statistical models have been proposed for that purpose; for instance, normal distributions are used in conjunction with the Mahalanobis distance [4] or a mixture of Gaussian models is considered to account for multi-value pixels located on image edges or belonging to shadow regions [5]–[7]. Such models can also evolve with time to manage varying background characteristics [4], [8], [9]. These background subtraction methods have been widely used in the area of real-time segmentation, although they require a learning step to obtain knowledge of the background color distribution.

In addition to these models, graph cut methods have also been widely used to enforce smoothness constraints over image regions. After the seminal work of Boykov and Jolly [10], many approaches have followed that direction. For example, GrabCut [11] takes advantage of iterative optimization to reduce the user interaction required to achieve good segmentation. Li *et al.* proposed a coarse to fine approach in Lazy Snapping [12] that provides a user interface for boundary editing. Shape prior information are considered in [13], [14] to reduce segmentation errors in areas where both the foreground and background have similar intensities. Background cut [15] also reduces segmentation errors due to background clutter by exploiting color gradient information. Recently, graph cut based approaches have also been proposed for object segmentation in videos [16], [17]. Algorithms have been proposed to reduce the amount of user interactions by using only a few seed pixels to estimate object boundaries [18], [19]. These methods have demonstrated their abilities to extract foreground objects both in static images and video sequences. However, they usually require user interaction that can be significant according to the complexity of the images being processed and the expected quality of the results.

The aforementioned approaches assume a monocular context and do not consider multi-camera cues, even when available. However, foreground regions in several images of the same scene should correspond to the same 3D space region. In other words, foreground regions over different viewpoints should exhibit spatial coherence in the form of a common 3D space region. Early attempts in that direction were made in [20], [21] where depth information obtained from stereo images are combined to photometric information to segment foreground and background regions. Recently, Kolmogorov *et al.* in [22] also proposed a real-time segmentation method that preserves the foreground object boundaries, under background changes, by combining stereo and color information. Incorporating depth information clearly improves over monocular cues when segmenting foreground objects. Nevertheless such approaches are designed for stereo imaging systems and do not easily extend to multi-camera systems with more than 2 cameras.

For more than 2 view configurations, spatial coherence is advantageously considered through a spatial region instead of locally through pixel depths. Again, consistent foreground image regions give rise to a single 3D space region. Conversely this region should project entirely on foreground regions in image domains, otherwise it would mean that there are space regions that correspond to foreground with respect to some viewpoints, and background with others. A few approaches exploit such fact through various scenarios. Zeng and Quan [23] proposed a method that propagates color consistency between viewpoints by iteratively carving the visual hull with respect to color consistency in each image. This approach increases spatial consistency from one to another viewpoint, however it only approximates spatial coherence which should be enforced over all viewpoints simultaneously.

In another work, Sormann *et al.* [24] applied a graph cut method to the multi-view segmentation problem. Spatial coherence is enforced over different viewpoints by minimizing differences between silhouette regions in 2 images at successive iterations. Such *shape prior* is combined to color information to segment shape silhouettes in multiple views. While improving over monocular approaches, this scheme relies on a strong assumption, i.e., silhouette similarities between two neighboring views, that is hardly satisfied even with small camera motion between two images. Bray *et al.* made use of *shape priors* to solve for both segmentation and model poses simultaneously under assumption of a known shape model, e.g., an articulated model [25].

For unknown shapes, Campbell *et al.* [26] recently proposed a 3D object segmentation approach with objectives similar to ours. They exploit both color and silhouette coherence and solve for the optimal 3D segmentation using a volumetric graph cut method. However, the object of interest is assumed to be at the center of all images and the segmentation is achieved in an intermediate voxel grid while we focus on the original image pixels. Another interesting direction is the occupancy grids [27]–[29]. In that case, background models are assumed to be known and 2D probability maps are fused into a 3D occupancy grid. Again, 2D silhouettes are not directly estimated but obtained as a by-product of a 3D segmentation in the occupancy grid, hence attaching the 2D silhouette segmentation to an unnecessary 3D discretization.

Our primary motivation is to propose a method that automatically identifies foreground regions in several images without prior knowledge nor user interaction. Monocular segmentation based on color consistency of the background and foreground image regions, e.g., [11] and [12], are not sufficient with arbitrary images where strong gradients perturb the segmentation and require user interactions. Spatial consistency among multiple views helps in that respect by providing additional constraints for the segmentation. Instead of using an intermediate 3D grid to enforce such constraints as in [26], [28], we directly formulate spatial consistency in the pixel domain and combine the resulting constraints with color consistency constraints. In addition to maintain the segmentation as a 2D process, such strategy assumes color consistency within each view and not among them, hence removing the need for color calibration when multiple cameras are considered.

III. Probabilistic Model

The framework we propose relies on the identification of the relationships between the entities involved, namely pixel colors, foreground and background models and binary silhouette labels. These relationships can be modeled in terms of probabilistic dependencies from which we can infer silhouette probability maps, as well as foreground and background models, given the pixel observations. To this purpose, we borrow the formalism developed by Franco and Boyer [28] for 3D occupancy grids. Similarly to this work we assume that the image observations are explained by the knowledge of the background in 2D and

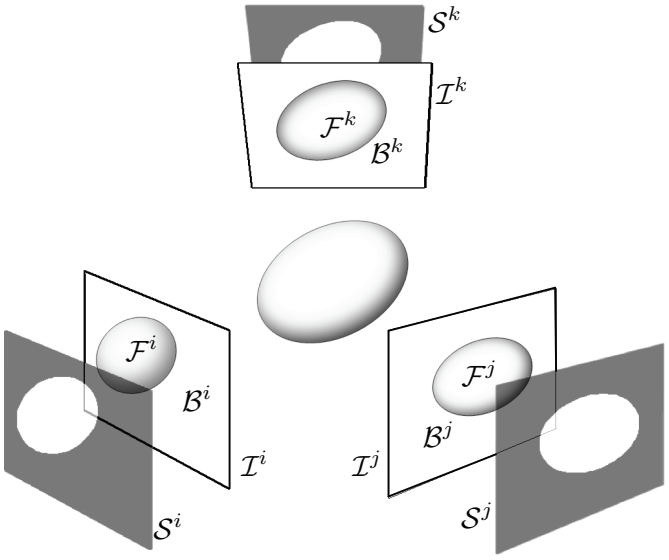


Fig. 2: The variables in different views.

by the 3D foreground occlusions (see Fig. 2 and 3). However, instead of explicitly modeling occupancy in 3D through a grid we define a shape prior that models the dependency between a pixel's occupancy in one image and pixel occupancies in all other images. Though similar in principle, the latter strategy is independent on any 3D discretization and allows us to directly solve for the pixel occupancies. The following sections detail the corresponding probabilistic modeling.

A. Variables and their Dependencies

Let us denote by $\mathcal{I}$ a color image map, by $\mathcal{S}$ a binary silhouette map, and by τ what is known beforehand about the model, e.g imaging parameters. Knowledge of the foreground occupancy and the background colors is denoted as $\mathcal{F}$ and $\mathcal{B}$, respectively. Note that $\mathcal{F}$, $\mathcal{B}$, and $\mathcal{S}$, are unknown variables while $\mathcal{I}$ is the only known variable in the problem. For each pixel, $\mathcal{S}$ has a value 0 if the pixel belongs to the background and 1 otherwise. We use the superscript i to represent a specific view, and the subscript $\mathbf{x}$ to indicate a pixel located at $\mathbf{x} = (u, v)$ in an image. Thus $\mathcal{I}^i_{\mathbf{x}}$ represents the color value of the pixel $\mathbf{x}$ in the i^{th} image. The variables $\mathcal{F}$, $\mathcal{B}$, $\mathcal{S}$, and $\mathcal{I}$ in different views are depicted in Fig. 2.

As shown in the dependency graph in Fig. 3, we assume that an image observation, $\mathcal{I}^i_{\mathbf{x}}$, is influenced by the background color at the corresponding pixel location, $\mathcal{B}^i_{\mathbf{x}}$, and by the fact that the background is occluded or not at that location, $\mathcal{S}^i_{\mathbf{x}}$, which is itself governed by the projection of the foreground region, $\mathcal{F}_{\mathbf{x}}$. We assume $\mathcal{F}$ and $\mathcal{B}$ to be independent which can be argued since shadows cast by the foreground can change the background appearance. However, and without loss of generality, we assume that shadows have a negligible impact on the background colors.

B. Joint Probability

Before we infer any probabilities from our Bayesian network, we need to compute the joint probability of all the variables. Using the dependency graph explicated in the previous section, we can decompose the joint probability $Pr(\mathcal{S}, \mathcal{F}, \mathcal{B}, \mathcal{I}, \tau)$ as:

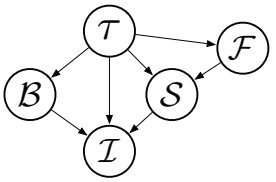


Fig. 3: Dependency graph of the image $\mathcal{I}$. $\mathcal{B}$ is the background color model, $\mathcal{S}$ the binary silhouette map, $\mathcal{F}$ the foreground spatial model and τ the prior knowledge about the model.

$$\begin{aligned} Pr(\mathcal{S}, \mathcal{F}, \mathcal{B}, \mathcal{I}, \tau) &= Pr(\tau)\, Pr(\mathcal{B}|\tau)\, Pr(\mathcal{F}|\tau) \\ &\quad Pr(\mathcal{S}|\mathcal{F}, \tau)\, Pr(\mathcal{I}|\mathcal{B}, \mathcal{S}, \tau). \end{aligned} \quad (1)$$

where:

- $Pr(\tau)$, $Pr(\mathcal{F}|\tau)$, and $Pr(\mathcal{B}|\tau)$ are the prior probabilities of the scene, foreground, and the background, respectively. Here, no *a priori* constraints are given on the background colors nor on the foreground shape. Thus, we assume they have uniform distributions and, as such, do not play any role in the inference.
- $Pr(\mathcal{S}|\mathcal{F}, \tau)$ is the silhouette likelihood that determines how likely is a silhouette given the foreground shape. Since $\mathcal{F}$ is unknown, and as explained below, we approximate this term by a spatial consistency term that determines how likely is a silhouette $\mathcal{S}^i$ given all the silhouettes $\mathcal{S}^{j\neq i}$.
- $Pr(\mathcal{I}|\mathcal{B}, \mathcal{S}, \tau)$ is the image likelihood term that models the relationship between the image observations, i.e., colors, and the background information.

Pixel measures, color or silhouette occlusion, can be assumed to be independent given their main causes namely background colors and foreground shape. Thus, the above distributions can be simplified to pixel term products as follows:

$$\begin{aligned} Pr(\mathcal{S}|\mathcal{F}, \tau) &= \prod_{i,\mathbf{x}} Pr\left(\mathcal{S}^i_{\mathbf{x}}|\mathcal{F}_{\mathbf{x}}, \tau\right), \\ Pr(\mathcal{I}|\mathcal{B}, \mathcal{S}, \tau) &= \prod_{i,\mathbf{x}} Pr\left(\mathcal{I}^i_{\mathbf{x}}|\mathcal{B}^i_{\mathbf{x}}, \mathcal{S}^i_{\mathbf{x}}, \tau\right). \end{aligned}$$

The above spatial consistency and image likelihood terms are detailed in the following sections.

C. Spatial Consistency Term

Silhouettes are the image regions onto which the foreground shape projects. The silhouette likelihood $Pr(\mathcal{S}|\mathcal{F}, \tau)$ is then the probability of a silhouette $\mathcal{S}$ knowing the foreground shape $\mathcal{F}$. Such a term reflects the fact that all silhouettes are generated by the same shape $\mathcal{F}$. Consequently, silhouettes from different viewpoints are not statistically independent unless the foreground shape is known. In fact silhouettes should be such that there exist a 3D region that projects onto all. This is known as the silhouette consistency constraint [30]. We exploit this property to constrain the shape of a silhouette given other silhouettes of the same 3D scene. The silhouette likelihood given the shape becomes therefore a spatial consistency term as follows:

$$Pr\left(\mathcal{S}^i_{\mathbf{x}}|\mathcal{F}, \tau\right) \simeq Pr\left(\mathcal{S}^i_{\mathbf{x}}|\mathcal{S}^{j\neq i}, \tau\right),$$

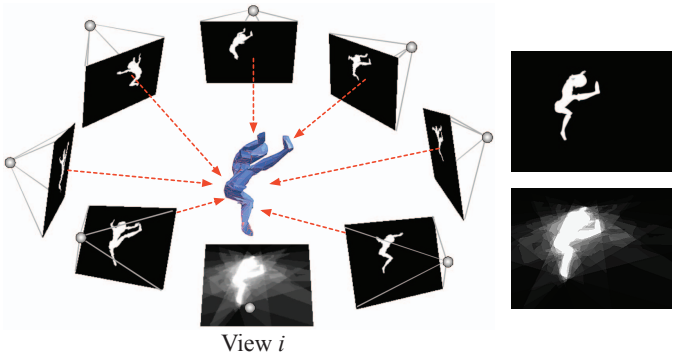


Fig. 4: The silhouette consistencies of pixels in image i: brighter pixels have higher consistencies. Top right: the true silhouette from viewpoint i; Bottom right: silhouette consistency measures of all pixel in image i given all silhouettes $\mathbf{S}^{j\neq i}$.

and to evaluate the silhouette consistency between viewpoints, we use the silhouette calibration ratio, introduced in [30], as explained below.

A set of silhouettes define a visual hull [31] which is the maximal volume consistent with all silhouettes. The visual hull is thus the intersection of the backprojection of silhouettes into 3D, i.e., the viewing cones. In a perfect world with exact silhouettes and calibration, a viewing ray from any pixel inside any silhouette intersects both the observed object and the visual hull and therefore all the other viewing cones [30]. The silhouette calibration ratio measures how true this property is for any pixel. It is a purely geometric measure that tells whether a pixel belongs to a silhouette according to the other silhouettes from different viewpoints and given the calibration. Figure 4 illustrates this principle and shows that silhouettes from viewpoints $j \neq i$ give a strong shape prior for the silhouette in image i.

As detailed in [30], the silhouette calibration ratio $C_{\mathbf{x}}$ at pixel $\mathbf{x}$ is a discrete measure based on the intersections between the viewing ray at $\mathbf{x}$ and the viewing cones from other viewpoints. In its simplest form, it takes values in the range $[0..N-1]$, where N is the number of views and $C^m = N-1$ denotes the highest consistency value. Assuming that $C_{\mathbf{x}}$ follows a normal distribution $R_{\mathbf{x}}$ below the true value C^m we have:

$$R_{\mathbf{x}} = \frac{1}{c} e^{-(C^m - C_{\mathbf{x}})^2/\sigma^2}, \tag{2}$$

where c is a normalization factor and σ controls how $C_{\mathbf{x}}$ influences the silhouette consistency term. In practice, σ reflects the confidence we have in silhouettes and should be chosen in order to allow for some tolerance. In our experiments, we typically use a value of 0.7 for σ.

We have defined the silhouette consistency term. We can now express the spatial consistency term at a given pixel location $\mathbf{x}$. The silhouette information at that pixel $\mathcal{S}^i_{\mathbf{x}}$ is a binary value: 0 for background and 1 for foreground. In the case where the pixel $\mathbf{x}$ is assumed to be background, i.e., $\mathcal{S}^i_{\mathbf{x}} = 0$, the silhouette information from other viewpoints does not provide any additional cue whether this is true or not. Hence, we assume the spatial consistency to follow a uniform distribution $\mathcal{P}_b$ in that case. On the other hand, when the pixel $\mathbf{x}$ is assumed to be foreground, i.e., $\mathcal{S}^i_{\mathbf{x}} = 1$, $R_{\mathbf{x}}$ tells us whether this is consistent with other silhouettes. Consequently, the spatial consistency term is as follows:

$$Pr\left(\mathcal{S}^i_{\mathbf{x}} | \mathcal{S}^{j\neq i}, \tau\right) = \begin{cases} \mathcal{P}_b & \text{if } \mathcal{S}^i_{\mathbf{x}} = 0, \quad (3a) \\ R_{\mathbf{x}} & \text{if } \mathcal{S}^i_{\mathbf{x}} = 1. \quad (3b) \end{cases}$$

D. Image Likelihood Term

The image likelihood term $Pr\left(\mathcal{I}^i_{\mathbf{x}} | \mathcal{B}^i, \mathcal{S}^i_{\mathbf{x}}, \tau\right)$ measures the similarity between a pixel color $\mathcal{I}^i_{\mathbf{x}}$ and the background information, i.e., the background color model at that location. In the same manner as for the spatial consistency term, there are 2 different situations. If a pixel belongs to the background, its color should follow the statistical color model of the background. Conversely, when the pixel is considered to be in the foreground region, the background color model does not provide any information about its color. As we make no assumptions regarding the color distribution of the foreground, we assume that the image likelihood term has a uniform distribution $\mathcal{P}_f$ in that case. Hence, the image likelihood term is defined as:

$$Pr\left(\mathcal{I}^i_{\mathbf{x}} | \mathcal{B}^i, \mathcal{S}^i_{\mathbf{x}}, \tau\right) = \begin{cases} \mathcal{H}_B\left(\mathcal{I}^i_{\mathbf{x}}\right) & \text{if } \mathcal{S}^i_{\mathbf{x}} = 0, \quad (4a) \\ \mathcal{P}_f & \text{if } \mathcal{S}^i_{\mathbf{x}} = 1, \quad (4b) \end{cases}$$

where $\mathcal{H}_B$ denotes the statistical model of the background colors. The value of $\mathcal{P}_f$ controls the threshold between foreground and background assignments and it ranges from 0 to 1. With large $\mathcal{P}_f$, pixels should have high likelihood to be classified as foreground, while pixels tend to be identified as background more easily with smaller $\mathcal{P}_f$. In practice, we set $\mathcal{P}_f$ to values specific to the data sets. Note however that in a more general approach, $\mathcal{P}_f$ can evolve during the iterative process, since $\mathcal{H}_B$'s evolves, by automatic thresholding as proposed in [32].

$\mathcal{H}_B$ can be estimated using several methods such as histograms or Gaussian mixture models and the overall approach we propose in this paper could consider any of them. In this work, we adopted a k component Gaussian mixture model (GMM). GMM have proven to be a powerful tool when solving segmentation problems [11], [26] and they are largely used for modeling color distributions. Using GMM, the image likelihood term is computed as the following sum of weighted probabilities:

$$\mathcal{H}_B\left(\mathcal{I}^i_{\mathbf{x}}\right) = \sum_k w_k \, \mathcal{N}\left(\mathcal{I}^i_{\mathbf{x}} | m_k, \Sigma_k\right) \tag{5}$$

where $\mathcal{N}(x|m_k, \Sigma_k)$ is the normal distribution with mean vector m_k and covariance matrix Σ_k. The value of k can vary depending on the application but a typical value used in our work is $k = 5$.

E. Inference of the Silhouettes

Once a joint probability distribution is defined, we can infer the silhouettes from the given conditions by exploiting Bayes rule. At pixel $\mathcal{I}^i_{\mathbf{x}}$, the probability of the silhouette is given by:

$$\begin{aligned} &Pr\left(\mathcal{S}^i_{\mathbf{x}} | \mathcal{S}^{j\neq i}, \mathcal{B}^i, \mathcal{I}^i_{\mathbf{x}}, \tau\right) \quad (6) \\ &= \frac{Pr\left(\mathcal{S}^i_{\mathbf{x}}, \mathcal{S}^{j\neq i}, \mathcal{B}^i, \mathcal{I}^i_{\mathbf{x}}, \tau\right)}{\sum_{\mathcal{S}^i_{\mathbf{x}}=0,1} Pr\left(\mathcal{S}^i_{\mathbf{x}}, \mathcal{S}^{j\neq i}, \mathcal{B}^i, \mathcal{I}^i_{\mathbf{x}}, \tau\right)}, \\ &= \frac{Pr\left(\mathcal{S}^i_{\mathbf{x}} | \mathcal{S}^{j\neq i}, \tau\right) Pr\left(\mathcal{I}^i_{\mathbf{x}} | \mathcal{B}^i, \mathcal{S}^i_{\mathbf{x}}, \tau\right)}{\sum_{\mathcal{S}^i_{\mathbf{x}}=0,1} Pr\left(\mathcal{S}^i_{\mathbf{x}} | \mathcal{S}^{j\neq i}, \tau\right) Pr\left(\mathcal{I}^i_{\mathbf{x}} | \mathcal{B}^i, \mathcal{S}^i_{\mathbf{x}}, \tau\right)}. \end{aligned}$$

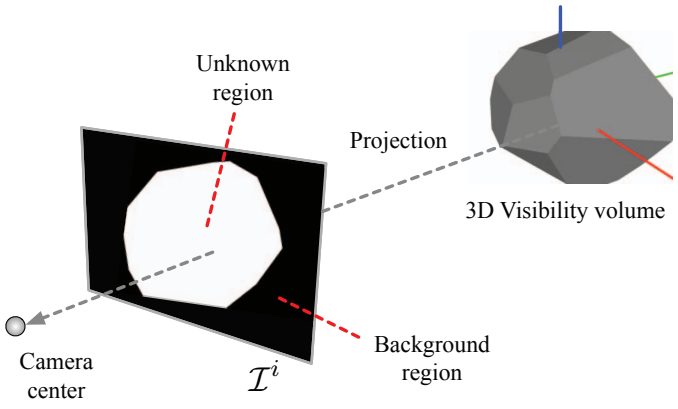


Fig. 5: Initialization: the initial silhouette of $\mathcal{I}^i$ is obtained by projecting the visibility volume onto $\mathcal{I}^i$. Note that the image region outside the silhouette necessarily belongs to the background while the initial silhouette contains both background and foreground elements.

The above expression allows the silhouette probability to be determined by combining both color information given by the background model and spatial constraints provided by other silhouettes. Applying it to a silhouette in a given image requires silhouettes in all other images to be known. This naturally leads to an iterative scheme where silhouettes are progressively improved by propagating silhouette shape constraints among viewpoints and updating background models accordingly.

IV. Iterative Silhouette Estimation

Our approach is grounded on the two assumptions which are frequently satisfied. First, any foreground element has an appearance different from the background in most images, so that color segmentation positively detects the element in most images. Second, we assume that the region of interest, i.e., the foreground, appears entirely in all the images considered. Hence, spatial consistency constraints hold since all foreground regions correspond to a single 3D space region. These two assumptions allow us to build initial models for the background and foreground which are then iteratively optimized in a two-step process: first silhouettes are estimated using foreground and background models, i.e., spatial and color consistencies, second these models are updated with the new silhouettes.

A. Initialization

We do not assume any prior knowledge on the background and foreground models. In order to initialize both models we use the fact that since the foreground scene is observed by all cameras, it necessarily belongs to the 3D space region that is visible from all cameras. Such a region is easily obtained as the visual hull of all 2D image domains, i.e., the 2D regions that occupy full images. When projected onto the image planes, this visibility volume defines initial foreground silhouettes. This is illustrated in Fig. 5 where the initial silhouette of $\mathcal{I}^i$ is obtained by projecting the visibility volume onto $\mathcal{I}^i$.

As shown in Fig. 5 the region outside the projected volume belongs to the *background*. We thus use the pixels in that region to initialize the background color model defined in section III-D.

B. Iterative Optimization via Graph Cut

The initialization described previously provides initial silhouettes as well as initial models for background regions. We then iterate the following 2 steps:

1) Estimate each silhouette $\mathcal{S}^i$ using (6) with the current background models $\mathcal{B}^i$ and the other current silhouettes $\mathcal{S}^{j\neq i}$.
2) Update each $\mathcal{B}^i$ with pixels outside the current $\mathcal{S}^i$.

The second step above simply consists in rebuilding the statistical background models with the additional pixels newly labelled as background. For the first step, (6) provides probabilities from which we need to decide for the pixel labelling into foreground or background in each image. Several approaches could be considered for that purpose, from locally thresholding the probability at each pixel to more global methods, such as graph based approaches which account for additional spatial coherence in the image. We use a graph cut approach [11], [33] which find the pixel assignment $\mathcal{S}^i$ that minimizes the following energy in image i[1]:

$$E_t(\mathcal{S}^i|\mathcal{S}^{j\neq i},\mathcal{B}^i,\mathcal{I}^i) = \sum_{\mathbf{x}\in\mathcal{I}^i} E_d\left(\mathcal{S}^i_{\mathbf{x}}|\mathcal{S}^{j\neq i},\mathcal{B}^i,\mathcal{I}^i_{\mathbf{x}}\right) + \sum_{\substack{(\mathbf{x},\mathbf{y})\in\mathcal{N}^i \\ \mathcal{S}^i_{\mathbf{x}}\neq\mathcal{S}^i_{\mathbf{y}}}} \lambda E_s\left(\mathcal{I}^i_{\mathbf{x}},\mathcal{I}^i_{\mathbf{y}}\right), \quad (7)$$

where:

- E_d is the data term that measures how good a pixel label $\mathcal{S}^i_{\mathbf{x}} = 0, 1$ is with respect to the image observation and for which we use the silhouette probability in (6) :

$$E_d\left(\mathcal{S}^i_{\mathbf{x}}|\mathcal{S}^{j\neq i},\mathcal{B}^i_{\mathbf{x}},\mathcal{I}^i_{\mathbf{x}}\right) = Pr\left(\mathcal{S}^i_{\mathbf{x}}|\mathcal{S}^{j\neq i},\mathcal{B}^i,\mathcal{I}^i_{\mathbf{x}},\tau\right).$$

- E_s is the smoothness term that favors consistent labelling in homogeneous region and $\mathcal{N}^i$ denotes the set of neighbouring pixel pairs in image i based on 8-connectivity.

$$E_s\left(\mathcal{I}^i_{\mathbf{x}},\mathcal{I}^i_{\mathbf{y}}\right) = \frac{1}{1+D\left(\mathcal{I}^i_{\mathbf{x}},\mathcal{I}^i_{\mathbf{y}}\right)},$$

where $D()$ is the Euclidean distance. Such energy penalizes neighbouring pixels with similar colors but different labels. It can take different forms as proposed in [11], [33] with similar results according to our experiments.

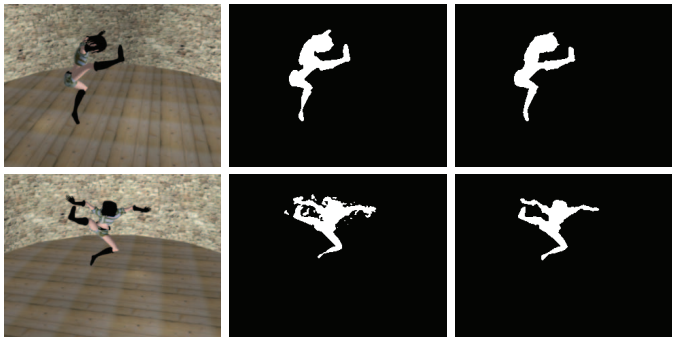
The graph cut approach finds new silhouette labels from which new background models are inferred before next iteration. To terminate the iterative optimization, we observe the number of pixels whose states changed from *Unknown* to *Background* and stop the process when no further pixels are newly identified as being in the background.

C. Silhouette refinement

The iterative scheme described in the previous section efficiently discriminates background and foreground pixels when there are either color cues with respect to background models or spatial cues with respect to other silhouettes. In some cases, in particular with few viewpoints, ambiguities remain because spatially consistent 3D regions project onto regions for which color information is not sufficient to correctly label. This is

[1]Note that a global minimization over all images cannot be considered here since to compute the spatial consistency term of the silhouette probabilities in a given image labels in all other images are required

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.



(a) (b) (c)

Fig. 6: Silhouette refinement: (a) input images; (b) silhouettes after the iterative optimization; (c) silhouettes after refinements.

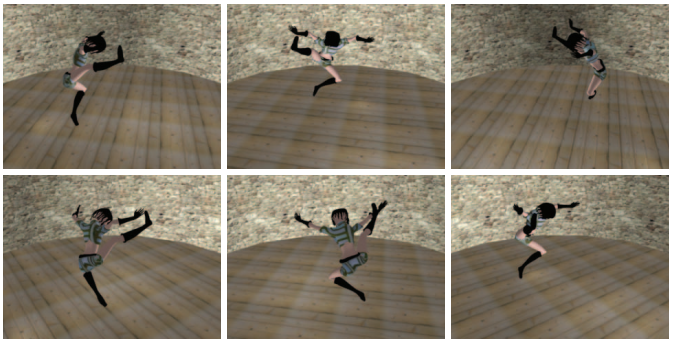


Fig. 7: The 6 input images of the *Kung-fu Girl* sequence.

typically the case nearby foreground object boundaries (see Fig. 6). Such ambiguities can be resolved by either adding viewpoints, thus refining the spatial consistency term, or by adding color information. We consider the latter in practice since the number of viewpoints is generally fixed. To this purpose, we make the assumption that the iterative optimization provides reasonable approximations of foreground regions, i.e., they contain a majority of foreground pixels. Under this assumption, we can build color models $\mathcal{H}_F$ for foreground regions to replace the uniform distribution $\mathcal{P}_f$ in the image likelihood term (**??**) which becomes:

$$Pr\left(\mathcal{I}^i_{\mathbf{x}}|\mathcal{B}^i,\mathcal{S}^i_{\mathbf{x}},\tau\right) = \begin{cases} \mathcal{H}_B\left(\mathcal{I}^i_{\mathbf{x}}\right) & \text{if } \mathcal{S}^i_{\mathbf{x}}=0 \quad (8a) \\ \mathcal{H}_F\left(\mathcal{I}^i_{\mathbf{x}}\right) & \text{if } \mathcal{S}^i_{\mathbf{x}}=1. \quad (8b) \end{cases}$$

To estimate $\mathcal{H}_F$, we use the GMM method presented in section III-D for $\mathcal{H}_B$ and then perform a graph cut step as described previously. Fig. 6 illustrates that approach with a synthetic example. Before refinement, in column (b), it can be seen that the silhouettes have both over- and under-estimated regions meaning that during the iterative optimization some foreground regions were lost while some background regions were not removed. As shown in column (c), the refinement with a non-uniform foreground color model significantly improves the results.

V. Experimental Results

In order to evaluate the proposed scheme, experiments with both synthetic and real data sets were performed. Standard real data sets, such as the Middlebury data set [41], were considered to demonstrate the interest of the approach in classical situations. In addition to real data sets, a synthetic data set was used to illustrate the behavior of the approach with challenging background and foreground color ambiguities.

A. Implementation

Experiments were performed on a 2.4GHz PC with 2GB RAM. The smoothing coefficient in the graph cut step was set to $\lambda = 1.2$. The uniform probability of a background pixel to be spatially consistent was set to $\mathcal{P}_b = 0.4$ and $\mathcal{P}_f$ varies depending on data sets[2]. These parameters were experimentally determined in this work. The experiments show that most of the processing time is devoted to the spatial consistency. Computing the spatial consistency term for a pixel requires projecting the viewing line of that pixel in all available images [30], thus the complexity is linear in the number of images for a pixel. Since all pixels in all images are considered, the overall complexity is $O(N_i^2\ N_p)$, where N_i is the number of images and N_p the number of pixels per image, and computation time for spatial consistency is typically several minutes for 8 images with 640×480 pixels without any implementation optimization. This can be drastically reduced by considering spatial consistency only at pixels which do not present high background probabilities at the previous iteration. In addition, it should be noted that the spatial consistency computation could easily be parallelized since computations are performed per pixel independently.

B. Synthetic data

We used the publicly available *Kung-fu Girl* sequence [42]. The data set consists of 25 calibrated images of a synthetic scene. For the experiments 6 views were selected, as shown in Fig. 7.

To illustrate the interest of the spatial consistency term for silhouette extraction, experiments where spatial consistency is enforced over different number of images, from 1 to 6, were conducted. In Fig. 8, the silhouettes (top row) and the corresponding spatial consistencies (bottom row) are shown. In all experiments, the background model was initialized with pixels outside the visibility volume of the 6 views, as described in section IV-A. In the single view case, the spatial consistency is not defined, thus all pixels are assumed to be consistent, i.e., the left image in Fig. 8. In that case, only background color consistency holds hence giving poor segmentation results since color information is not discriminant enough for this data set. As the number of views increases, more background regions are progressively identified. This shows that although background and foreground colors are similar, the spatial consistency provides useful cues that can disambiguate the segmentation.

Fig. 9 shows the segmentation results obtained using the proposed method. Since the cameras have symmetric poses, the initial silhouettes are almost identical, as illustrated in the second row. The next rows $3-6$ show the segmentation results at different iterations. Note that even with a challenging situation where foreground and background colors present similarities, the foreground regions can still be automatically identified with a reasonable precision. In addition, though parts of the foreground can be lost during optimization, most are recoverable through the post-processing step by exploiting the foreground color model, as shown in row 7.

[2]In this work, we used the following values for $\mathcal{P}_f$: $\mathcal{P}_f = 0.65$ for *Dancer*, *Temple*, *Toy-1*, and *Duck-1*; $\mathcal{P}_f = 0.7$ for *Toy-2*, *Duck-2* and *Violet*; $\mathcal{P}_f = 0.75$ for *Kung-fu Girl* and *Bust*

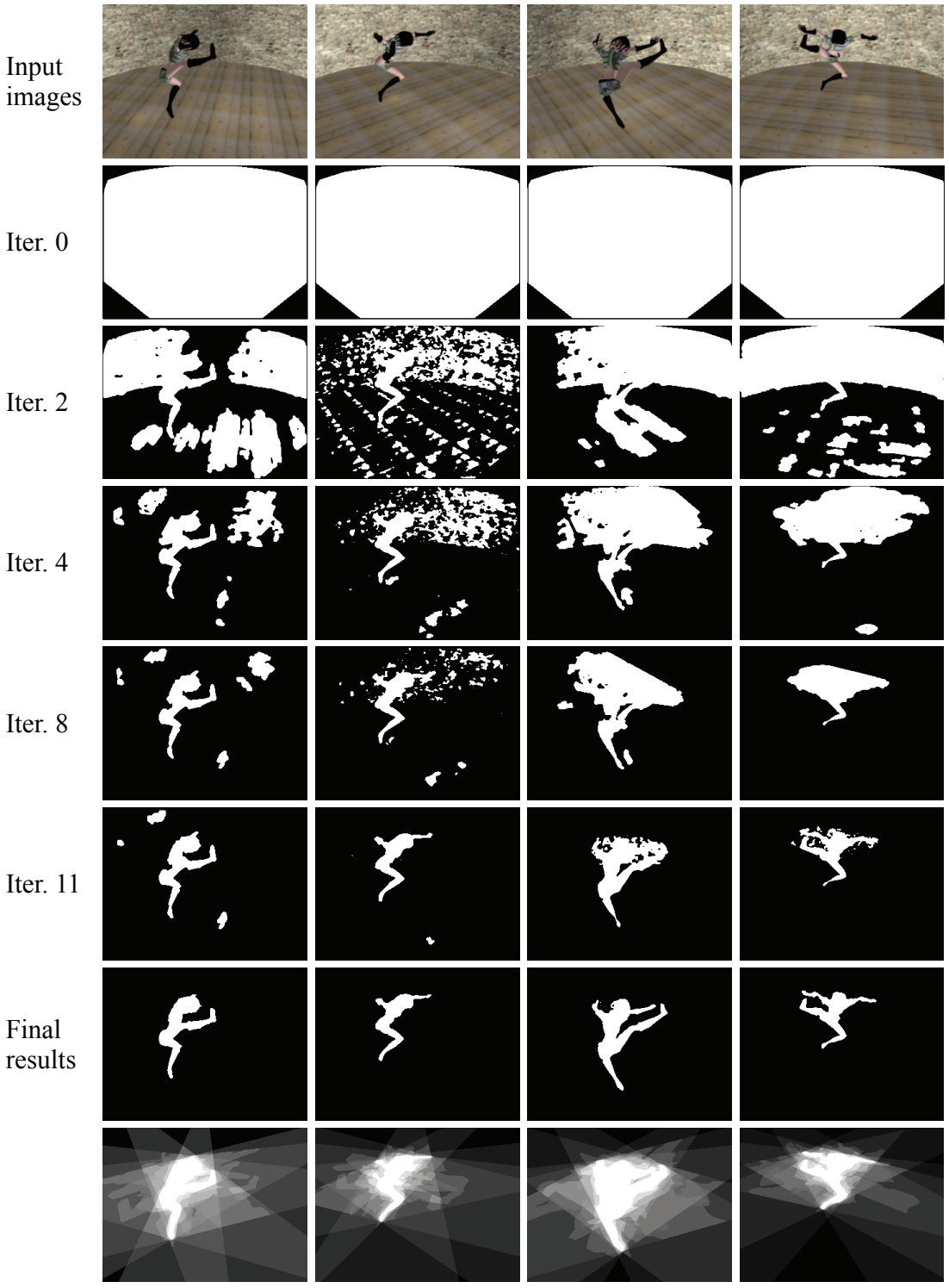


Fig. 9: Segmentation results with the *Kung-fu Girl* sequence. Top row: input color images; row 2: initial segmentation obtained by projecting the camera visibility domain; rows $3-6$: segmentation results at different iterations; row 7: final segmentations after post-processing, bottom row: spatial consistencies corresponding to the final segmentation.

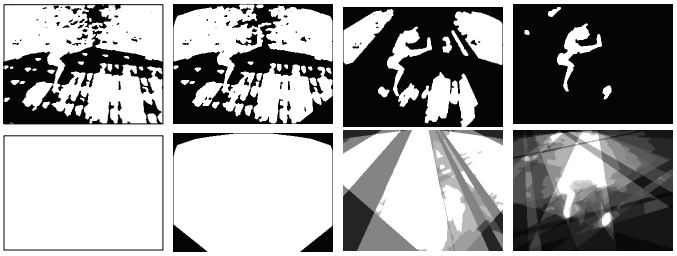


Fig. 8: Segmentation results with different number of views (top row) accounting for spatial consistencies (bottom row). From left to right, 1, 2, 4, and 6 views are used. Note that segmentation errors that occur with 6 views are due to color similarities between background and foreground regions. Such artefacts will generally be removed with a post-processing step, as explained in section IV-C.

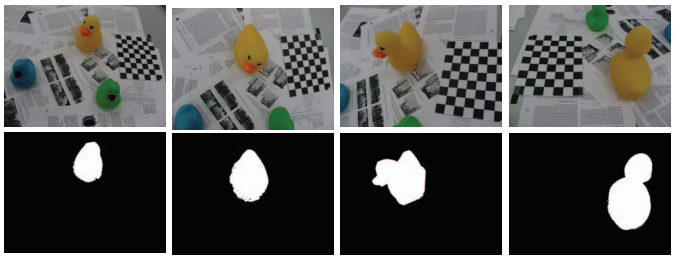
C. Real data

In order to evaluate the approach in practical situations, several multi-view data sets were considered. These sets were captured both under controlled-lighting conditions and under general lighting conditions. Note that color calibration was not performed with the data sets used in these experiments. In the following, we first explain how camera calibration is conducted then show the results of silhouette segmentation.

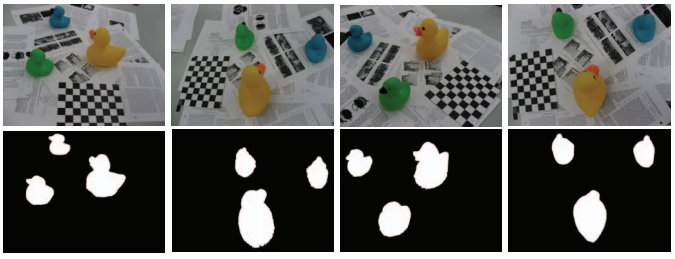
The images are used for our experiment are calibrated as follows. For simple data sets, we used a checkerboard pattern, which is a well-known basic calibration method, and many implementations are available [46], [47]. For data sets of complex scenes, we follow the structure from motion technique for camera motion estimation. First, we extract SIFT [34] features from all images. We adopt GPU-based implementation to improve the speed of feature extraction [48]. Then, we find the two images with the highest feature similarity among all input images. Using these two selected images, a two-view reconstruction is carried out to obtain an initial set of sparse 3D points followed by a bundle adjustment. The camera pose is initialized using Nister's 5 points algorithm [35]. After the two-view reconstruction, we incrementally add remaining images to the reconstruction. This approach returns the camera poses with a reasonable accuracy. In the pose estimation step, we assume that the intrinsic parameters of the camera are known and that only the extrinsic parameters need to be estimated. For intrinsic parameters, retrieving CCD sensor information from the EXIF tags of images is one solution as proposed in [36]. Note that some of the data sets used in our experiments are already calibrated.

In Fig. 10, silhouette extraction results obtained with the *Dancer* data sequence [43] are shown. They illustrate that precise silhouettes can be extracted in real situations without prior information on the background and with the sole assumption that foreground objects appear in all images. We show more experimental results in Fig. 11 with data sets having simple and complex backgrounds.

The *Temple* data set [41] presents an almost uniformly black background, making therefore the silhouette extraction easier than in other cases. Nevertheless, note that, as illustrated in the table I below, the temple belongs to the foreground region but presents colors similar to the foreground making the object boundaries difficult to extract precisely. The *Toy-1* data set corresponds to a typical setting for image-based modeling where the background has colors different from the foreground. The *Duck-1* sequence illustrates a more complex situation with non-uniform background and strong edges in the images. The *Toy-2* and *Duck-2* data sets present more complex backgrounds. The *Bust* [44] and *Violet* [45] data sets also present complex and natural scenes although *Violet* contains both simple and complex backgrounds depending on the viewpoint. In all data sets, the lighting conditions differ with respect to the viewpoints. Hence, color consistency cannot be assumed between viewpoints while geometric consistency still holds. As shown in Fig. 11, our approach extracted the silhouettes of foreground object successfully from both simple and complex scenes. Interestingly in the *Duck* data set, the checkerboard pattern is identified as background. This is explained by the fact that its colors belong to the background model and also because it is not fully spatially consistent since some parts do not project inside all images, thus contradicting the two assumptions of our approach. We can see a similar situation in the *Bust* data set results, where only the statue is identified as foreground although the wooden support is visible in all views. This is because the legs of the support are clipped in some views, making the support part spatially inconsistent. In the results with the *Violet* data set, some details of small stems are lost but the overall object shape is well retrieved.



(a)



(b)

Fig. 12: Silhouette extraction with multiple object scenes. In (a) 1 object only is spatially consistent and in (b) all the three objects are spatially consistent. Both data sets consist of 6 views.

Fig. 12 presents experimental results with data sets where multiple objects are observed. In Fig. 12(a), only 1 object is identified as foreground as a result of the spatial consistency assumption that foreground objects appear in all images. Thanks to our spatial consistency constraint, the small ducks' beaks are identified as foreground although their color belongs to the background model. In contrast, in Fig. 12(b), all objects are correctly extracted in the images showing that the algorithm correctly identifies the foreground region seen by all images without supervision, i.e., without the need for specific information about its content.

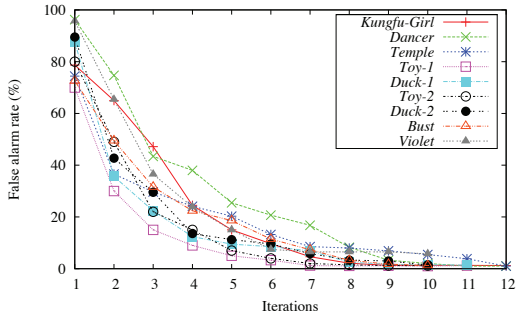


Fig. 13: Convergence of the extracted silhouettes: the average false alarm rates at each iteration.

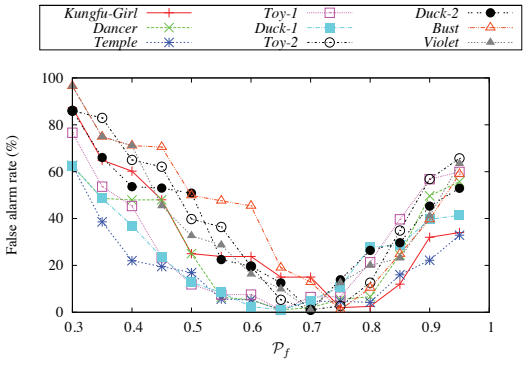


Fig. 14: Silhouette extraction with different $\mathcal{P}_f$: large and small values of $\mathcal{P}_f$ increase the false detection rate because most pixels are identified as either foreground or background in that case.

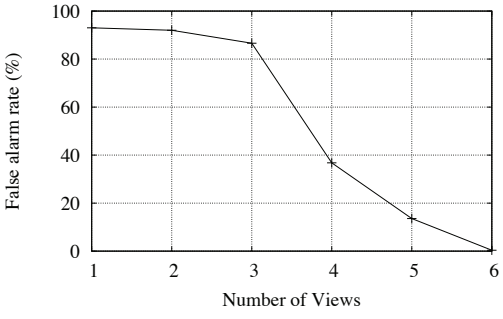


Fig. 15: Silhouette extraction for different number of views

1) Quantitative evaluation: In the following we present a set of numerical evaluations that illustrates how the approach behaves with different data sets, over iterations and in the presence of noise. Ground truth silhouettes were obtained manually with the help of commercial software such as Photoshop or Gimp.

To compare the silhouettes obtained by our method and the ground truth, we denote by W_a^b the label set a pixel belongs to, where a is the labelling F or B obtained with our method and b the ground truth label. From these 4 sets of pixels, we can compute the rates of pixels correctly and incorrectly labelled foreground over foreground and background regions, respectively as:

$$\begin{aligned} \textit{Hit Rate} &= \frac{N\left(W_F^F\right)}{N\left(W_F^F\right)+N\left(W_B^F\right)} \\ \textit{False Alarm Rate} &= \frac{N\left(W_F^B\right)}{N\left(W_F^F\right)+N\left(W_B^B\right)}, \end{aligned} \tag{9}$$

where $N(\cdot)$ represents the number of pixels in a set. Such rates are then averaged over the different images.

Table I shows the results. Interestingly, the results with the synthetic data set are worse than with real data. This is mainly due to the strong ambiguities between foreground and background colors in the synthetic images. Our approach keeps high accuracy of the resulting silhouettes even with complex background although simple scene cases show more accurate results. The *Violet* sequence for instance shows less accuracy because of the small details lost as showed in Fig. 11. Note also that the highest standard deviation among the simple scene data sets, with the *Duck-1* sequence, results from the scale variations between viewpoints.

The behavior over iterations is illustrated in Fig. 13 for the different scenes. It shows that the false alarm rates decreases dramatically between iteration 1 and 8, as large areas of the background regions are removed at each iteration through the combination of color and spatial consistency constraints.

In the experiments, we manually chose $\mathcal{P}_f$ for each data set. As explained in Section III-D, pixels are more likely to be classified as foreground with larger $\mathcal{P}_f$, while smaller $\mathcal{P}_f$ increases pixels' likelihood to be identified as background. Fig. 14 illustrates such behavior with various values for $\mathcal{P}_f$. Results show that the data sets having simple backgrounds present a better tolerance to false foreground detection than data sets having complex backgrounds.

Another observation from Fig. 14 is that the false alarm rate is less than 100 although $\mathcal{P}_f$ is close to 1. This means that not all pixels are classified as foreground even with large $\mathcal{P}_f$ and demonstrates that the spatial consistency constraint can identify background regions although the foreground likelihood is high.

To evaluate how the number of views affect to silhouette segmentation results, we conducted silhouette extraction with varying number of views, ranging from 1 to 6. We used the *Kung-fu Girl* sequence for experiment and the result is shown in Fig. 15. As expected, more views result in better silhouette estimation (also illustrated in Fig. 8). It can also be seen that performances increase drastically with 4 views or more. The reason for that is because the spatial consistency becomes inaccurate with less than 4 views.

In order to measure the robustness of the proposed silhouette extraction method with respect to noise in the image pixel colors and in the calibration parameters, multi-view silhouette extraction was performed with varying noise levels on the *Kung-fu Girl* sequence. The averaged false alarm rate is depicted in Fig. 16. Pixel color noises were generated as random Gaussian noises, with zero means and standard deviations σ, which were added to all color channels in all images. For camera parameters, i.e., the focal length and translation parameters, the noise varies from 0% to 5% of the exact parameter values, and for rotation parameters the noise varies from 0 to 2 degrees in rotation angles with respect to the x, y, and z axes. Each point in the graphs corresponds to the mean value over 15 trials, obtained with a randomly chosen image frame from the full sequence of the *Kung-fu Girl* data set (i.e., 200

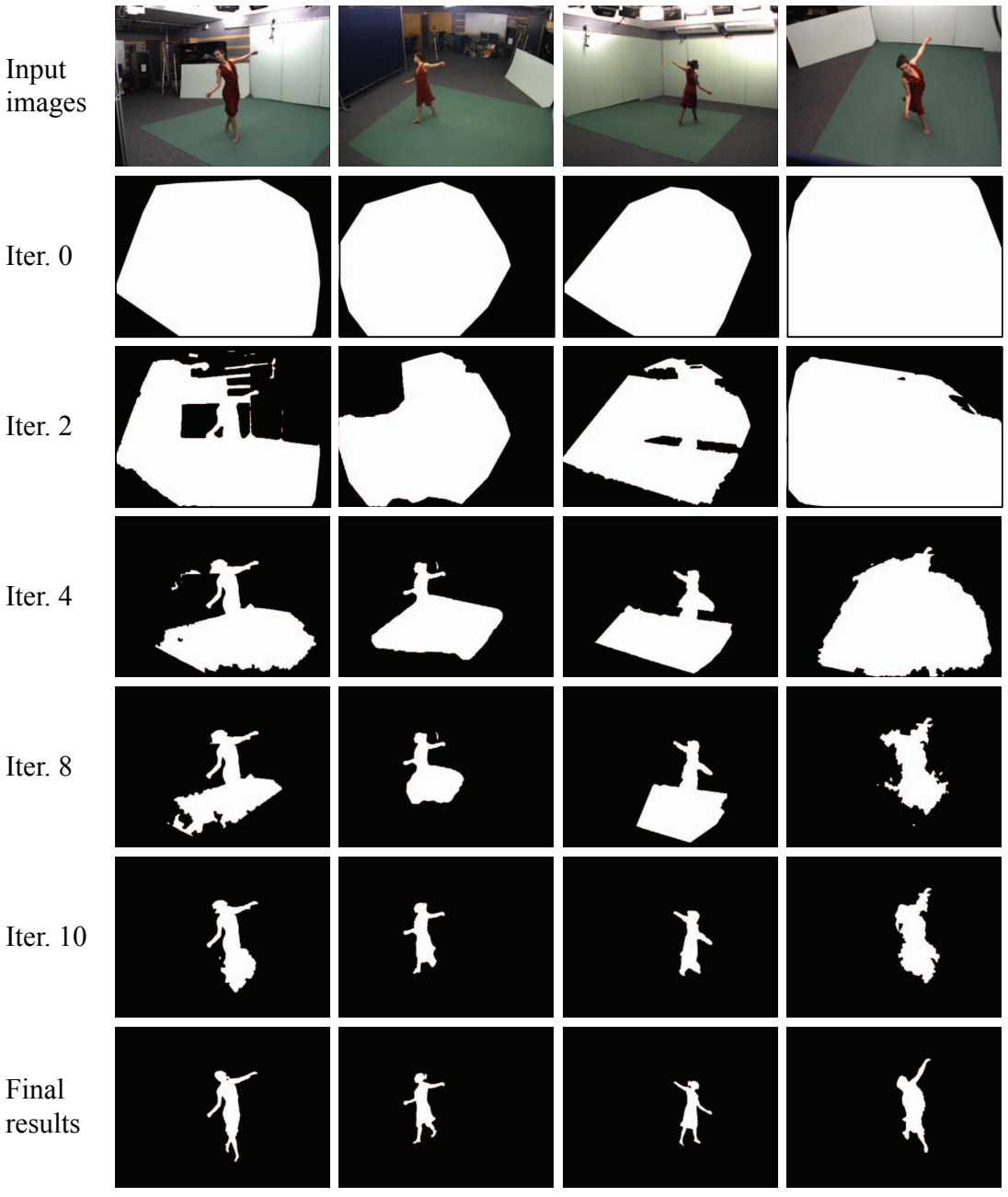
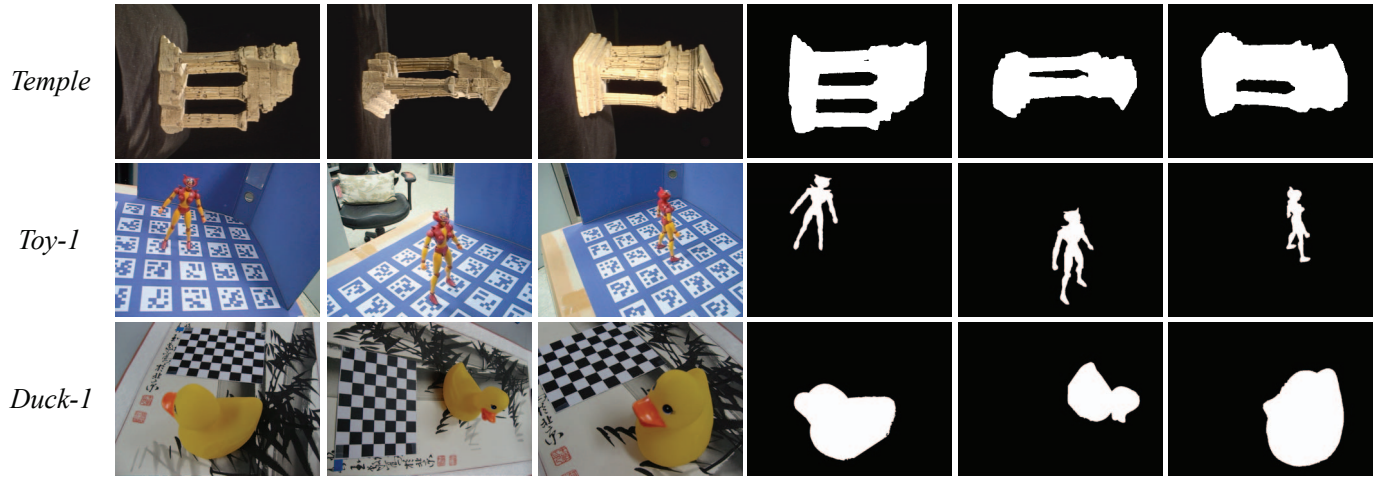


Fig. 10: Segmentation results with the *Dancer* data set (8 views). Top row: 4 selected images; row 2: initial silhouettes; rows $3-6$: segmentation results at different iterations; row 7: final segmentation after post-processing.

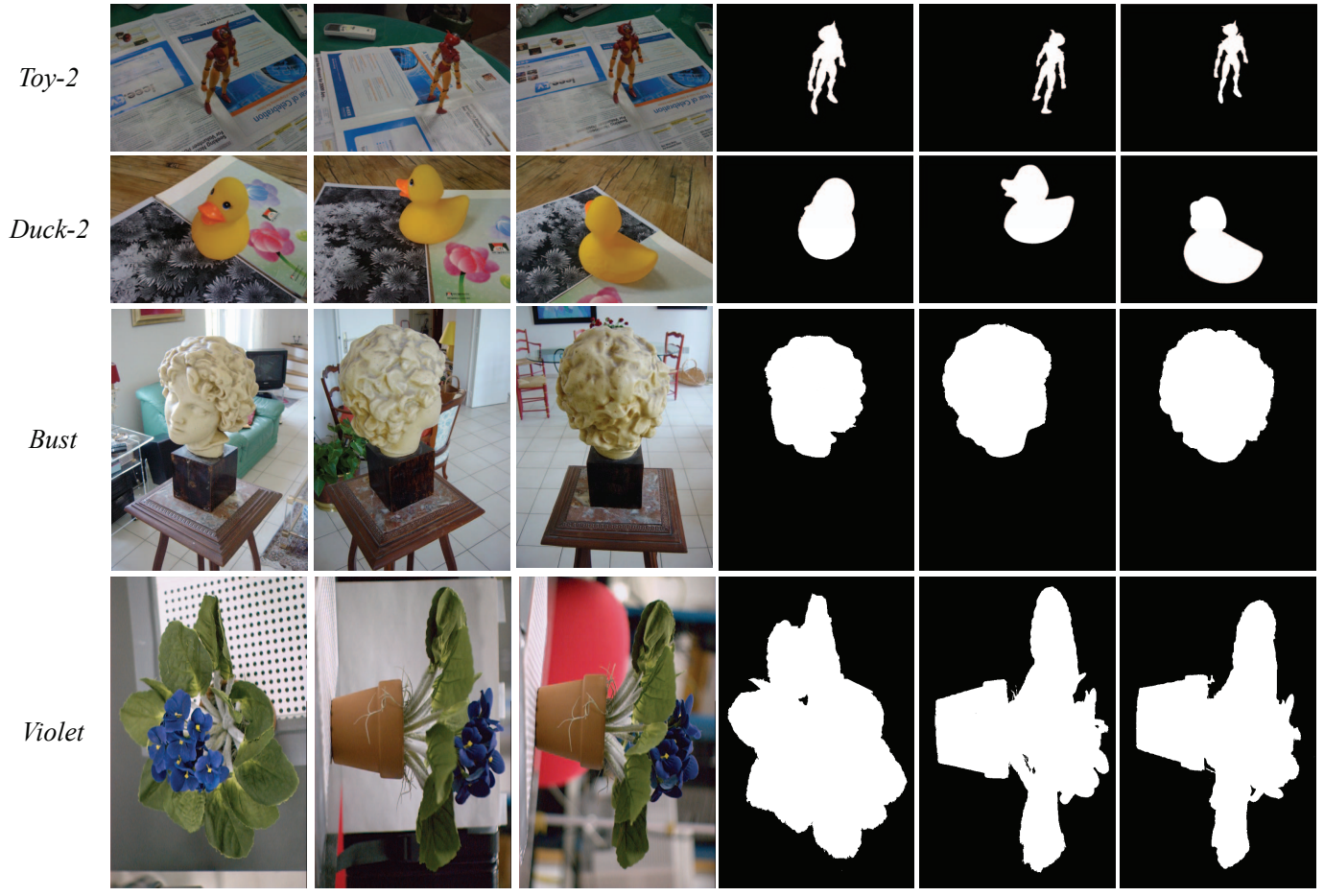
TABLE I: Silhouette extraction performance measurements.

	Hit Rate (%)		False Alarm Rate (%)	
	Mean	STD	Mean	STD
Kung-fu Girl (6 views)	84.66	4.1	2.1	1.17
Dancer (8 views)	94.45	1.46	1.79	0.24
Temple (10 views)	98.27	0.22	1.15	0.43
Toy-1 (12 views)	99.81	0.19	1.08	0.18
Duck-1 (5 views)	99.57	0.45	1.25	0.88
Toy-2 (12 views)	98.53	0.42	1.19	0.39
Duck-2 (8 views)	99.27	0.31	1.42	0.48
Bust (6 views)	98.94	0.82	1.25	1.32
Violet (6 views)	92.14	0.72	5.74	1.95

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.



(a)



(b)

Fig. 11: Silhouette extraction with single object scenes. (a) Results with a simple scene: from top to bottom, *Temple* (10 views), *Toy-1* (12 views), and *Duck-1* (5 views) (b) Results with more complex scene: from top to bottom, *Toy-2* (12 views), *Duck-2* (8 views), *Bust* (6 views), and *Violet* (6 views).

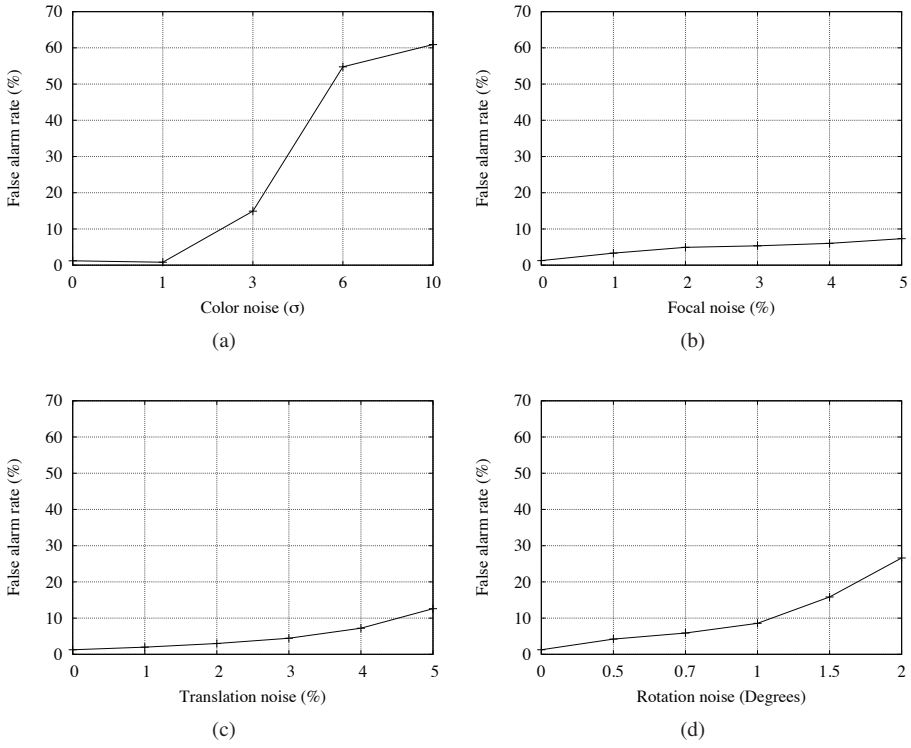


Fig. 16: Silhouette extraction in the presence of noise in (a) the pixel colors, (b) the focal lengths, (c) the translation parameters and (d) the rotation parameters.

frames). As shown in Fig. 16(a), the proposed method is robust to color noises with $\sigma \leq 3$, but the performances decrease drastically when $\sigma > 3$. Such behavior is, in part, due to the fact that noises modify colors in both background and foreground regions and that, in such a situation, the background and foreground color models are ambiguous and result in inaccurate classification results. With incorrect calibration parameters, the foreground regions inferred from other views may provide inaccurate spatial consistency cues. Hence, parts of the foreground regions are lost in the extracted silhouettes. According to our experimental results, the spatial consistency is more sensitive to errors in the rotation parameters than errors in the translation or focal length parameters. Also, these results show that the approach is more sensitive to errors in colors than errors in spatial camera poses.

D. Discussion

1) Failure Cases: Although it shows good performances in our experiments, the proposed approach fails when the initial assumptions are not satisfied.

1) Color models of the foreground and background are indistinguishable due to similar color distributions or large color noises. As showed in Fig. 16, color noises can result1 in large errors.
2) Parts of the foreground object are clipped in some views. In that case, the clipped parts of the object do not satisfy spatial consistency and thus, they are likely to be identified as background.

A potential solution to these problems is to use a local color classifier for a better color consistency check and to apply an adaptive weighting scheme for the color and spatial consistencies as proposed in [37] for instance.

2) Limitations: The approach also presents some limitations. First, segmentation is difficult in the vicinity of object boundaries where colors are ambiguous. Such ambiguities occur during the image acquisition and are caused by the reflections of foreground colors onto background surfaces, and vice versa. Since spatial consistency is not necessarily very accurate in such regions, they can be therefore misclassified. This limitation can be overcome by exploiting other post-processing methods such as active contour [38] or by allowing some user interactions [24]. A second limitation comes from the fact that all images should be calibrated. This limitation can be addressed by a robust structure from motion algorithm that provides reconstruction of cameras from a set of unorganized images [36]. Another possible solution is exploiting a homographic framework for spatial consistency inference as proposed in [39], [40]. On the other hand, wrong calibration parameters penalize the spatial consistency term which becomes unreliable. A possible solution would be to simultaneously optimize calibration parameters in the process of estimating silhouettes.

VI. Conclusions

In this paper, we have presented a novel method for extracting spatially consistent silhouettes of foreground objects from several viewpoints. The method integrates both spatial consistency and color consistency constraints in order to identify silhouettes with unknown backgrounds. It does not require *a priori* knowledge on the scene nor user interaction and, as such, provides an efficient automatic solution to silhouette segmentation. The only assumptions made are that foreground objects are seen by all images and

that they present color differences with the background regions. Geometric constraints are enforced among viewpoints and color constraints inside each viewpoint. Results demonstrate the interest of the approach in practical configurations where 3D models are built using images from different viewpoints.

ACKNOWLEDGMENT

This research is supported by Ministry of culture, Sports and Tourism (MCST) and Korea Creative Content Agency (KOCCA), under the Culture Technology(CT) Research & Development Program 2010.

REFERENCES

[1] C. Hernández and F. Schmitt, "Silhouette and Stereo Fusion for 3D Object Modeling," *Computer Vision and Image Understanding*, vol. 96, no. 3, pp. 367–392, December 2004.

[2] Y. Furukawa and J. Ponce, "Carved Visual Hulls for Image-Based Modeling," in *European Conference on Computer Vision*, 2006, pp. 564–577.

[3] A. Zaharescu, E. Boyer, and R. Horaud, "Transformesh: a topology-adaptive mesh-based approach to surface evolution," in *Asian Conference on Computer Vision*, vol. LNCS 4844, 2007, pp. 166–175.

[4] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland, "Pfinder: Real-Time Tracking of the Human Body," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 780–785, 1997.

[5] S. Rowe and A. Blake, "Statistical mosaics for tracking," *Image and Vision Computing*, vol. 14, pp. 549–564, 1996.

[6] N. Friedman and S. Russell, "Image Segmentation in Video Sequences: A Probabilistic Approach," in *13th Conf. on Uncertainty in Artificial Intelligence*, 1997.

[7] C. Stauffer and W. Grimson, "Adaptative Background Mixture Models for Real-Time Tracking," in *IEEE Conference on Computer Vision and Pattern Recognition*, 1999, pp. 246–252.

[8] K. Toyama, J. Krumm, B. Brumitt, and B. Meyers, "Wallflower: Principles and practice of background maintenance," in *International Conference on Computer Vision*, September 1999, pp. 255–261.

[9] A. M. Elgammal, D. Harwood, and L. S. Davis, "Non-parametric Model for Background Subtraction," in *European Conference on Computer Vision*, 2000, pp. 751–767.

[10] Y. Boykov and M.-P. Jolly, "Interactive Graph Cuts for Optimal Boundary and Region Segmentation of Objects in N-D Images," in *International Conference on Computer Vision*, vol. 1, 2001, pp. 105–112.

[11] C. Rother, V. Kolmogorov, and A. Blake, "GrabCut-Interactive Goreground Extraction using Iterated Graph Cuts," in *ACM SIGGRAPH*, vol. 24, no. 3, 2004, pp. 309–314.

[12] Y. Li, J. Sun, C.-K. Tang, and H.-Y. Shum, "Lazy Snapping," in *ACM SIGGRAPH*, vol. 23, no. 3, 2004, pp. 303–308.

[13] D. Freedman and T. Zhang, "Interactive Graph Cut Based Segmentation With Shape Priors," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, June 2005, pp. 755–762.

[14] N. Vu and B. Manjunath, "Shape Prior Segmentation of Multiple Objects with Graph Cuts," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2008, pp. 1–8.

[15] J. Sun, W. Zhang, X. Tang, and H.-Y. Shum, "Background Cut," in *European Conference on Computer Vision*, 2006, pp. 628–641.

[16] Y. Li, J. Sun, and H.-Y. Shum, "Video object cut and paste," in *ACM Transactions on Graphics*, vol. 24, no. 3, 2005, pp. 595–600.

[17] J. Wang, P. Bhat, A. Colburn, M. Agrawala, and M. Cohen, "Interactive Video Cutout," in *ACM SIGGRAPH*, 2005, pp. 585–594.

[18] B. Micusik and A. Hanbury, "Automatic Image Segmentation by Positioning a Seed," in *European Conference on Computer Vision*, vol. LNCS 3952, 2006, pp. 468–480.

[19] E. N. Mortensen and J. Jia, "Real-Time Semi-Automatic Segmentation Using a Bayesian Network," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2006, pp. 1007–1014.

[20] I. Kompatsiaris, D. Tzovaras, and M. G. Strintzis, "3D Model-based Segmentation of Videoconference Image Sequences," in *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 8, no. 5, 1998, pp. 547–561.

[21] G. Gordon, T. Darrell, M. Harville, and J. Woodfill, "Background Estimation and Removal Based on Range and Color," in *IEEE Conference on Computer Vision and Pattern Recognition*, 1999, pp. 459–464.

[22] V. Kolmogorov, A. Criminisi, A. Blake, G. Cross, and C. Rother, "Probabilistic fusion of stereo with color and contrast for bi-layer segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 9, pp. 1480–1492, September 2006.

[23] G. Zeng and L. Quan, "Silhouette Extraction from Multiple Images of An Unknown Background," in *Asian Conference on Computer Vision*, vol. 2, 2004, pp. 628–633.

[24] M. Sormann, C. Zach, and K. Karner, "Graph cut based multiple view segmentation for 3d reconstruction," in *International Symposium on 3D Data Processing, Visualization and Transmission*, 2006.

[25] M. Bray, P. Kohli, and P. H. Torr, "PoseCut: Simultaneous Segmentation and 3D Pose Estimation of Humans using Dynamic Graph-Cuts," in *European Conference on Computer Vision*, 2006, pp. 642–655.

[26] N. Campbell, G. Vogiatzis, C. Hernández, and R. Cipolla, "Automatic 3D Object Segmentation in Multiple Views using Volumetric Graph-Cuts," in *British Machine Vision Conference*, vol. 1, 2007, pp. 530–539.

[27] D. Snow, P. Viola, and R. Zabih, "Exact voxel occupancy with graph cuts," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2000, pp. 345–352.

[28] J.-S. Franco and E. Boyer, "Fusion of Multi-View Silhouette Cues Using a Space Occupancy Grid," in *International Conference on Computer Vision*, 2005, pp. 1747–1753.

[29] L. Guan, J.-S. Franco, and M. Pollefeys, "Multi-object shape estimation and tracking from silhouette cues," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[30] E. Boyer, "On Using Silhouettes for Camera Calibration," in *Asian Conference on Computer Vision*, January 2006, pp. 1–10.

[31] A. Laurentini, "The visual hull concept for silhouette-based image understanding," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 2, pp. 150–162, Februrary 1994.

[32] N. Kim, W. Woo, G. Kim, and C.-M. Park, "3-D Virtual Studio for Natural Inter-Acting," *IEEE Transactions on Systems, Man and Cybernetics, Part A*, vol. 36, no. 4, pp. 758–773, July 2006.

[33] Y. Boykov, O. Veksler, and R. Zabih, "Fast Approximate Energy Minimization via Graph Cuts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 11, pp. 1222–1239, 2001.

[34] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[35] D. Nistér, "An efficient solution to the five-point relative pose problem," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 6, pp. 756–777, June 2004.

[36] N. Snavely, S. M. Seitz, and R. Szeliski, "Modeling the world from Internet photo collections," *International Journal of Computer Vision*, vol. 80, no. 2, pp. 189–210, November 2008.

[37] X. Bai, J. Wang, D. Simons, and G. Sapiro, "Video snapcut: robust video object cutout using localized classifiers," *ACM Transactions on Graphics*, vol. 28, no. 3, pp. 1–11, 2009.

[38] C. Xu and J. Prince, "Snakes, Shapes, and Gradient Vector Flow," *IEEE Transactions on Image Processing*, vol. 7, no. 3, pp. 359–369, 1998.

[39] S. M. Khan and M. Shah, "Reconstructing non-stationary articulated objects in monocular video using silhouette information," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[40] P.-L. Lai and A. Yilmaz, "Efficient object shape recovery via slicing planes," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–6.

[41] "*Temple* data set," Multi-view stereo evaluation web page. http://vision.middlebury.edu/mview/.

[42] "*Kung-Fu Girl* data set," http://www.mpi-inf.mpg.de/departments/irg3/kungfu/.

[43] "*Dancer* data set," Multiple-Camera/Multiple-Video Database. https://charibdis.inrialpes.fr/html/index.php.

[44] "*Bust* data set," http://www.cs.ust.hk/ quan/WebPami/pami.html.

[45] "*Violet* data set," Multi-View Stereo Datasets. http://www.cs.toronto.edu/ kyros/soft-data/static/index.html.

[46] "GML C++ Camera Calibration Toolbox," http://graphics.cs.msu.ru/en/science/research/calibration/cpp, 2009.

[47] "Camera Calibration Toolbox for Matlab," http://www.vision.caltech.edu/bouguetj/calib_doc, 2009.

[48] C. Wu, "SiftGPU: A GPU implementation of scale invariant feature transform (SIFT)," http://cs.unc.edu/ ccwu/siftgpu, 2007.

Wonwoo Lee received his B.S. in Dept. of Mechanical Engineering from Hanyang University, Seoul, S. Korea in 2003 and M.S. in the Dept. of Information and communication from Gwangju Institute of Science and Technology (GIST), Gwangju, S. Korea, in 2004. Currently, he is a Ph.D. student in DIC, GIST. His research interests include 3D computer vision, computer graphics, and augmented reality.

Woontack Woo received his B.S. in Electronics Engineering from Kyungpook National University in 1989 and his M.S. in Electronics and Electrical Engineering from POSTECH in 1991. In 1998, he received his Ph.D. in Electrical Engineering Systems from University of Southern California (USC). In 1999, as an invited researcher, he joined Advanced Telecommunications Research (ATR), Kyoto, Japan. Since Feb. 2001, he has been with the Gwangju Institute of Science and Technology (GIST), where he is an Associate Professor in the Department of Information and Communications and Director of Culture Technology Institute. His research interests include 3D computer vision and its applications including attentive AR and mediated reality, HCI, affective sensing and context-aware for ubiquitous computing, etc.

Edmond Boyer is senior researcher at the INRIA Rhne-Alpes (France) in the field of computer vision. He obtained his PhD from the Institut National Polytechnique de Lorraine (France) in 1996 and started his professional career as a research assistant at the University of Cambridge (UK) in the Department of Engineering. From 1998 to 2010, Edmond Boyer was associate professor at Grenoble universities (France) and researcher at the INRIA Rhône-Alpes. His fields of competence cover computer vision, computational geometry and virtual reality. He is co-founder of the 4D View Solution Company in the domain of multi-camera acquisition. His current research interests are on 3D dynamic modeling from images and videos, motion capture and recognition from videos, and immersive and interactive environments.