Depth-Color Fusion Strategy for 3-D Scene Modeling With Kinect

Massimo Camplani, *Member, IEEE,* Tomás Mantecón, and Luis Salgado, *Member, IEEE*

***Abstract*—Low-cost depth cameras, such as Microsoft Kinect, have completely changed the world of human–computer interaction through controller-free gaming applications. Depth data provided by the Kinect sensor presents several noise-related problems that have to be tackled to improve the accuracy of the depth data, thus obtaining more reliable game control platforms and broadening its applicability. In this paper, we present a depth-color fusion strategy for 3-D modeling of indoor scenes with Kinect. Accurate depth and color models of the background elements are iteratively built, and used to detect moving objects in the scene. Kinect depth data is processed with an innovative adaptive joint-bilateral filter that efficiently combines depth and color by analyzing an edge-uncertainty map and the detected foreground regions. Results show that the proposed approach efficiently tackles main Kinect data problems: distance-dependent depth maps, spatial noise, and temporal random fluctuations are dramatically reduced; objects depth boundaries are refined, and nonmeasured depth pixels are interpolated. Moreover, a robust depth and color background model and accurate moving objects silhouette are generated.**

***Index Terms*—3-D scene modeling, adaptive bilateral filter, data fusion, depth map filtering, Kinect, mixture of Gaussians.**

I. Introduction

THREE-DIMENSIONAL content generation is becoming a fundamental issue in many multimedia applications, being particularly challenging when interaction between users and virtual objects and scenarios is required to provide new experiences. The great success of low-cost depth cameras with good resolution and acquisition frame rate, such as the Microsoft Kinect [1], is capturing the attention of the game industry, developers, and research community; thus, multiplying the number of proposed 3-D-based applications. In the field of computer games, low-cost depth cameras have been widely employed in controller-free video games. The captured depth data is used to extract the user silhouette, track its body parts, such as hands and head [2], or the entire set of skeleton joints [3], and then interpret user movements and gestures as game commands. For these purposes, several natural user interface (NUI) toolkits have been proposed such as [4]. Other attractive applications are immersive environments [5] where different users silhouette are projected in a shared virtual space where they can interact.

However, depth data provided by low-cost depth cameras is in general seriously affected by different types and levels of noise that limit the game control platforms, robustness, and accuracy. Innovative filtering strategies are required to efficiently improve depth map's quality and stability. This way, the performance of body parts,tracking algorithms, user segmentation strategies, or 3-D modeling approaches can be improved, thus, raising control game usability while boosting the quality of the experience in case of, for example, shared virtual scenarios. In the recent survey on depth-based gesture recognition and body tracking algorithms [6], has highlighted the importance of new and efficient algorithms that improves depth data accuracy provided by the new generation of depth cameras. For example, the performance of the gesture recognition system presented in [7] is boosted by using a filtered depth map.

Microsoft Kinect is a device composed by a color and a depth sensor based on a structured light 3-D scanner. An infrared light source projects light patterns to the space: the reflected light is received by an infrared camera and compared with a reference pattern produced by a plane at a known distance from the camera. The differences between the acquired patterns and the reference ones allow to estimate a disparity image of the scene, and hence to extract the depth measurements. Structured light sensor measurements are affected by noise due to the multiple reflections, transparent objects, or scattering in particular surfaces (i.e., human tissue and hair) and strong lighting conditions.

In Fig. 1 the color image (a) and the corresponding depth map (b) (pixel depth values are grey level coded) of an indoor environment obtained with the Kinect are shown. As it can be observed, one of the main problems of the Kinect depth map is the presence of pixels (marked in red) for which no depth data is provided. These nonmeasured depth pixels (*nmd* pixels) are mainly due to occlusions (typically around object boundaries) or to scattering of particular surfaces, but they also appear in areas that correspond to concave surfaces (i.e., empty spaces in the library) and, randomly, in homogeneous image regions (i.e., above the black door on the right side of the image).

Manuscript received April 9, 2012; revised November 23, 2012 and May 10, 2013; accepted June 13, 2013. Date of publication July 16, 2013; date of current version November 18, 2013. This work was supported in part by the Ministerio de Economia e Competitividad of the Spanish Government under Project TEC2010-20412, Enhanced 3-DTV. The work of M. Camplani was supported by the European Union and the Universidad Politécnica de Madrid through the Marie Curie Co-Fund Research Grant. Paper recommended by Associate Editor V. Argyriou.

M. Camplani and T. Mantecón are with the Grupo de Tratamiento de Imágenes, UPM, Madrid 28040, Spain (e-mail: mac@gti.ssr.upm.es; tmv@gti.ssr.upm.es).

L. Salgado is with the Video Processing and Understanding Laboratory, Universidad Autónoma de Madrid, Madrid 28049, Spain, and also with the Grupo de Tratamiento de Imágenes, ETSI Telecomunicación, Universidad Politécnica de Madrid, Madrid 28040, Spain (e-mail: l.salgado@gti.ssr.upm.es).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCYB.2013.2271112

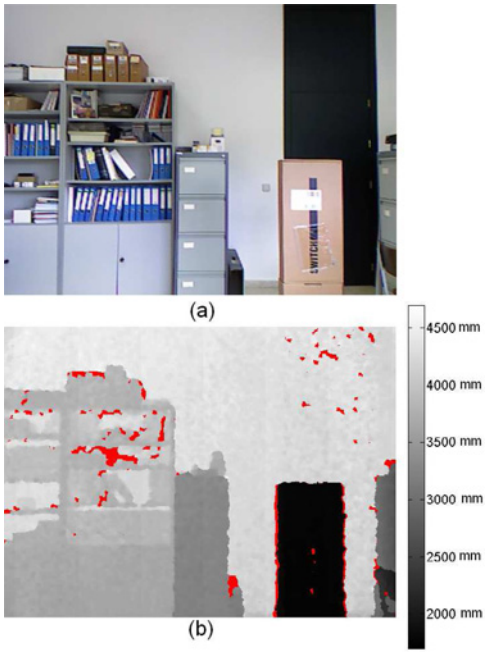

Fig. 1. (a) Kinect color data. (b) Corresponding depth map where pixel depth values are grey level coded.

Depth measurements at object boundaries are also heavily affected by noise. Sharp depth transitions produce misleading reflection patterns that result in rough and inaccurate depth measurements that are far from being correctly aligned to the actual object boundaries. This can be observed in Fig. 1(b), for example, at the boundaries of the brown rectangular box closer to the camera (gray-level close to black).

Depth measurements are also affected by instability over time and space. On one side, measurements taken for a static object that correspond to the same image pixel vary with time. On the other side, different depth values are obtained for spatially neighboring pixels that correspond to points situated at the same distance from the camera. Moreover, the impact of this error varies with the distance; in fact, as shown in [8], the theoretical dispersion of depth measurements varies with the distance following a quadratic law. This variation pattern has been confirmed by our tests in which the camera was placed orthogonally to a white wall, and sequences of 200 frames were taken at distances ranging from 0.6 to 5 m. In Fig. 2 (dashed line) the standard deviation, σ_{noise}, of the measured depth is reported as a function of the wall–camera distance. As expected, σ_{noise} increases with the distance following a quadratic function (solid line). More information about Kinect depth measurements can be found in [8] and [9].

In this paper, we present a depth-color fusion strategy for 3-D modeling of indoor scenes with Kinect that efficiently tackles these problems and, therefore, improves the accuracy of the Kinect depth maps. This paper is based on some of the concepts developed in [10] and [11].

The proposed method iteratively builds accurate depth-based and color-based models of the scene background (composed by quasi-static scene elements). These models are used to detect moving elements in the scene. Depth data provided by the Kinect is processed with an innovative

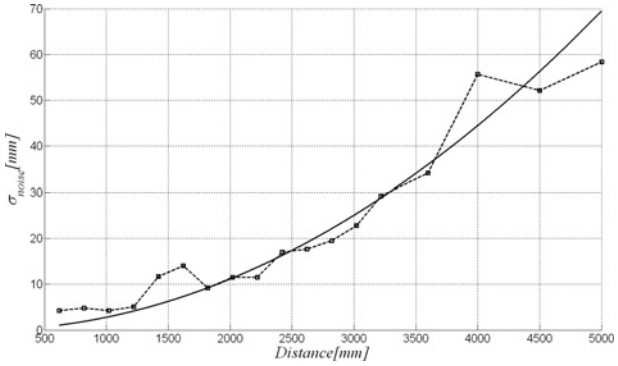

Fig. 2. Depth measurements dispersion as function of the object–camera distance.

adaptive joint-bilateral filter that efficiently combines depth and color data by analyzing an edge-uncertainty map and the detected foreground regions. The depth model data and the color information are used to reduce the *nmd* pixels. The innovative features of the proposed approach are: the adaptive filter parameters, selected considering the distance dependent depth-map noise; the use of the edge-uncertainty map for color and depth fusion and the combination of the adaptive depth and color models to efficiently manage scenarios that include also moving objects. Results demonstrate that the proposed approach efficiently reduces main Kinect depth data problems while building a robust background model, and generates accurate moving objects silhouette.

The paper is structured as follows: in Section II an overview of Kinect denoising approaches is given; in Section III the proposed strategy is presented; Section IV contains the results and in Section V the conclusions are drawn.

II. Related Work

Recently, several Kinect-depth-based applications have been presented in different computer vision areas; however, there are few works that deal with the problem of Kinect depth data improvement. Other works in literature focus on the depth map processing obtained with stereo and time-of-flight (ToF) devices, but their proposals are not suitable to be directly employed to solve Kinect depth data problems. In this section, we will present some of the most significant works available in the literature related to Kinect depth data processing.

The approach proposed in [12] is based on the analysis of motion information (motion vectors) to register images in time, incorporating a noncausal spatio-temporal median filtering to account for depth data noise. Although good results are reported for some image areas, the strategy is not able to solve the problems of noisy object boundaries, and computational requirements prevent it from being used in real-time applications.

In [13], a GPU-based filtering system for the Kinect depth map is presented. The *nmd* pixels are restored by using the normalized convolution and the rest of the depth map is filtered using the guided filter. Although operating in real time, limited depth map improvement is obtained mainly because only depth information is considered in the filtering process: erroneous depth values are interpolated from invalid depth

measurements, and the objects noisy borders in depth are preserved and eventually blurred.

In the telepresence system proposed in [5], the noisy Kinect data are processed by a cascade of two different filters. In the first stage, a median filter is used to preserve objects' depth boundaries while reducing spatial noise. The same median filter is used to remove *nmd* pixels that have a neighborhood with a sufficient number of valid depth measurements. The obtained depth map is then processed by a simple bilateral filter applied to the obtained depth data to reduce the flattering effect introduced by the median filtering. The filters used in this approach do not consider the color information, hence leading to wrong *nmd* interpolated values. Moreover, the final bilateral step does not guarantee object boundaries refinement.

Regarding the *nmd* pixels interpolation, a recursive median filter is proposed in [14] and used to generate an object recognition dataset. However, none of the other Kinect data problems is addressed to improve the depth data accuracy.

The problem of *nmd* pixels interpolation is also faced in [15], where an extension of the fast marching-based image inpainting method [16] is proposed. In particular, the authors fuse in the weighting function spatial relationship, depth data of the *nmd* neighborhood and structural components coherence computed on the color image. The main disadvantages of this paper are that the temporal information is not considered in the interpolation process, and that the interpolated *nmd* pixels belonging to object boundaries are not properly refined. Finally, the complexity of this algorithm is superior with respect to the other interpolation approaches.

Although each of the above-mentioned works address individual problems of the Kinect depth data, they show a limited exploitation of color data, temporal consistency, and lack adaptability, especially, to the depth data noise model. The particularly challenging refinement at depth object boundaries is barely formally approached, and quantitative evaluation of results is scarce; therefore, there are several advantages of using the proposed method proposed. First, its adaptation capabilities: filtering parameters are dynamically adapted according to the distance-dependent noise model, the reliability of depth information and the dynamic or static nature of the scene elements. Second, the use of a robust temporal model to efficiently reduce depth temporal fluctuations, and iteratively build a robust model of quasi-static scene background. Finally, the effective joint exploitation of color and depth data, that is fundamental to improve the depth-map quality at object boundaries and properly interpolate *nmd* pixels.

III. System Overview

The proposed system efficiently combines depth and color information provided by the Kinect to reduce the noise present in the depth maps and to improve their accuracy. In particular, it allows reducing the distance-dependent spatial noise and the *nmd* pixel effect while accurately preserving object depth boundaries, thanks to an adaptive filtering strategy. Temporal fluctuations of the depth measurements are reduced by iteratively building a reliable color/depth model of the static elements in the scene. The parameters of the proposed filtering process and the temporal model are continuously adapted to the distance-dependent noise.

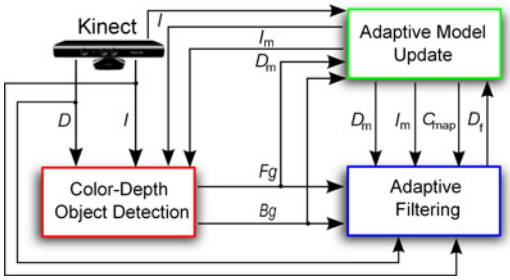

Fig. 3. Block diagram of proposed system.

The proposed system is composed by three main modules as reported in Fig. 3: the color-depth object detection module (CDObj), the adaptive filtering (AF) module, and the adaptive model update (AMU) module.

CDObj combines the background color-based model I_m and depth-based model D_m to detect moving objects in the scene. The different dynamics of the scene elements, quasi-static for the background elements (Bg) and moving for the foreground ones (Fg), are in fact considered in the filtering process.

The innovative AF, built on the basis of a joint-bilateral filter [17], improves the accuracy of each acquired depth map D considering the dynamic of the scene (represented by Bg and Fg), and fusing the acquired color image information I and the background color-based model I_m, with the depth-based model D_m. A measure of the depth-based model reliability is contained in the consistency map C_{map}: an image where the greater the value of a pixel, the more reliable is its corresponding depth value in D_m. This map is used in the AF to drive the *nmd* replacement strategy where only the most reliable depth measurements are taken into account.

Finally, in AMU, the background depth-based model D_m and color-based model I_m are iteratively built and updated. The acquired color image I and the obtained filtered depth map D_f are used together with Fg. For those pixels in D_f belonging to the background, the corresponding C_{map} value is also updated. Both models, D_m and I_m, are based on the mixture of Gaussian (MOG) background modeling algorithm [18]. In the following sections, the main features and design concepts of these modules are thoroughly presented.

It is worth noting that the entire system aims at a local enhancement of the depth data, combining at the pixel level a spatial and temporal processing. This pixel-wise approach is very attractive for any real-time application, such as controller-free video games or human–computer interaction systems, since it guarantees a feasible parallel implementation for example using GPU architectures.

A. Color-Depth Object Detection: CDObj

The proposed strategy adapts the filtering parameters to the static or dynamic nature of the elements in the scene; thus, it requires accurate moving object detections. The color-depth object detection module CDObj, presented graphically in Fig. 4, efficiently combines both models to precisely detecting the regions containing moving objects in the scene.

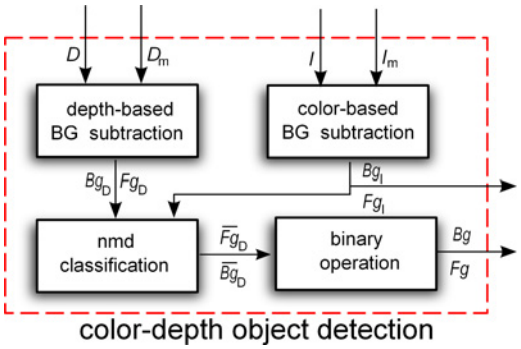

Fig. 4. Details of the color-depth object detection module.

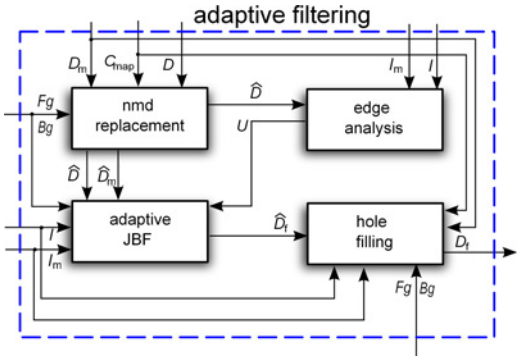

Fig. 5. Details of the adaptive filtering module.

Depth-based object detection binary mask Fg_D is computed: each pixel is classified as foreground or background considering D_m and the acquired depth map D. Correspondingly, the color-based object detection mask Fg_I is estimated considering I_m and I. Details of the statistical test used for pixel classification are given in Section III-C.

Depth-based detections result in compact silhouettes in Fg_D, not affected by illumination changes or shadows but showing, on the one hand, imprecise and noisy contours and, on the other hand, unclassified pixels (the *nmd* pixels in D). On the contrary, detections based on color information typically present accurate object contours in Fg_I, but may lack compactness (color camouflage, noise) and include false positives (illumination changes, shadows). Combining both types of information helps completing the depth-based detection map and improve detection accuracy, but require the identification of the image areas where color-based detected object contours should prevail over those in the depth-based detection. Binary image operations are proposed to combine depth and color-based detections, thus, rendering a binary mask Fg preserving depth-based compactness and color-based accuracy in the detections.

Color-based classification in Fg_I is used for the *nmd* pixels in Fg_D, thus, completing the depth-based detection mask $\overline{Fg}_D$. The depth-based detected objects boundary, severely affected by the noise in the depth-map D, is refined through morphological operations, and an improved foreground objects detection mask Fg is computed as

$$Fg = \left(Fg_I \cap \left(\overline{Fg}_D - \varepsilon_B\left(\overline{Fg}_D\right)\right)\right) \cup \varepsilon_B\left(\overline{Fg}_D\right) \qquad (1)$$

where $\varepsilon_B(\bullet)$ stands for the morphological erosion with a structuring element B. The corresponding background detection mask is the inverse of the foreground one: $Bg = \sim Fg$. It is worth noting that the proposed strategy to combine depth and color information to generate improved foreground detection masks does not depend on the particular background modeling strategy used. In fact, a different background modeling approach can be applied as it will only affect the background subtraction block.

B. Adaptive Filtering: AF

The adaptive filtering block AF is based on an innovative joint-bilateral filter that reduces the distance-dependent spatial noise of the acquired depth map while refining object boundaries and interpolating coherent depth values for the *nmd* pixels. The filter parameters are selected as a function of the measured depth to efficiently reduce the distance-dependent noise. Moreover, the color information, by definition of an edge-uncertainty map, is only selectively used around depth-based borders for their refinement. Finally, the same filtering framework is also extended to the reduction of the *nmd* pixels.

AF takes into account the presence of static and moving objects in the scene: the main idea is to use the more stable information provided by the models D_m and I_m to filter the pixels belonging to the background. On the contrary, to filter the foreground pixels, the just acquired information I and D is used. In the following paragraphs for the lack of space, we report only the filter equations for the Bg case and highlight the differences with the Fg one. The details of the AF module are shown in Fig. 5.

The first step of the proposed strategy is the *nmd* replacement: using the actual depth map D and the stable values of the depth-based model D_m, two completed (up to what is possible) depth maps $\widehat{D}$ and $\widehat{D}_m$ are generated to be further used in the filtering process. The *nmd* pixels in D, if classified as background, are replaced with their corresponding depth values in D_m. The substitution is performed only if the corresponding depth value in D_m is reliable. In fact, the depth map model has associated a consistency map C_{map} that indicates the reliability of the depth values in the model. Given a pixel q, the greater the value of C^q_{map}, the higher the reliability of D^q_m; this value is considered reliable when it is greater than a threshold c_{th}. In our implementation, C_{map} is computed as the occurrence number of depth measurements that belong to the background model. The role of the consistency map C_{map} and D_m is fundamental to improve the filter performance, especially, in the hole filling strategy: they guarantee stable and reliable neighborhoods to interpolate the depth values of the *nmd* pixels. It is worth noting that in the case that D_m contains *nmd* pixels, they are substituted by the corresponding pixels in D, so that it is guaranteed that both $\widehat{D}$ and $\widehat{D}_m$ have the same set of *nmd* pixels during the filtering process.

Due to the noisy object depth boundaries, we propose to filter these regions' pixels considering the color information. Color data corresponding to depth discontinuity regions helps selecting for the filtering phase only pixels that likely belong to the same object, thus, refining the depth map at object boundaries. On the contrary, to filter homogeneous depth regions only depth data is considered, thus, avoiding that high

textured areas (in color) introduce artifacts in the filtered depth map. Through the analysis of the gradient of the depth map $\widehat{D}$ and the color information (I and I_m), an edge-uncertainty map U is extracted. Its values determine which model (depth-based or color-based) has to be used to estimate the pixels similarity in the adaptive filtering. U is a binary mask that identifies the depth discontinuity regions that are characterized by a low color correlation. Let us define E_D as an edge map obtained by applying an edge detector algorithm to the depth map D. The U map is estimated as follows:

$$U = \delta_B[E_D \cap I_{\sigma_I}],\ I^p_{\sigma_I} = \begin{cases} 1 & \text{if } \sigma(I, \Omega^p) > \sigma_I \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

where I_{σ_I} is the low color correlation map, $\sigma(I, \Omega^p)$ is the standard deviation of the color data in the neighborhood Ω^p of pixel p, and $\delta_B(\bullet)$ stands for the morphological dilation with structuring element B. Our tests have shown that a circular structuring element of radius about 4–5 pixels is sufficient to include in the U map the noisy depth boundaries and the more resilient edges in the color domain. As edge detector we use the Sobel operator. The adaptive joint bilateral filter is then applied to remove spatial noise while refining the object boundaries in the depth map.

Let us define $\widehat{D}^p$ the pixel in the depth map $\widehat{D}$ at the position p, and $\widehat{D}^p_m$ the corresponding pixel in $\widehat{D}_m$. The filtered depth value obtained with the proposed joint-bilateral filter is

$$\begin{aligned} \widehat{D}^p_f \quad = \quad & \frac{\omega_1}{k^p} \sum_{q \in \Omega^p} \left[\widehat{D}^q f_1(p, q) g_1 \left(\left\| \widehat{D}^p_m - \widehat{D}^q_m \right\| \right) \right] \\ & + \omega_2 f_2 \left(\widehat{D}, g_2 \left(I^p, I_m \right), \Omega^p \right) \end{aligned} \tag{3}$$

where Ω^p is the neighborhood of the pixel at position p, $f_1(\bullet)$ and $f_2(\bullet)$ are the spatial terms of the joint bilateral filter and the functions $g_1(\bullet)$ and $g_2(\bullet)$ are the range terms of the filter that determine the weights of the filter $f_1(\bullet)$ and $f_2(\bullet)$ by measuring the pixels similarity. The binary weights ω_1 and ω_2 depend on the values of U and are defined such that $\omega_2 = \sim \omega_1$. In particular, if the filtered pixel at position p belongs to the uncertainty zone, the similarity range term is calculated considering the color data and ω_1 is set to 0; otherwise ω_1 is set to 1.

Let us consider the case ω_1 equal to one; in this case, the filtered pixel p is obtained applying the depth-based adaptive joint-bilateral term. In particular, the spatial term $f_1(\bullet)$ is a smoothing Gaussian function that considers the closeness of the pixels; its weights are further modified by the range term, a Gaussian function $g_1(\bullet)$, that considers the depth similarity and adapts to the distance-dependent noise model

$$g_1(x) = e^{-\frac{1}{2}\left(\frac{x}{\sigma_d}\right)^2} \tag{4}$$

where σ_d is selected according to the depth value $\widehat{D}^p_m$ following the value of σ_{noise} shown in Fig. 2. In this way, the different levels of noise present in the depth images are considered in the filtering process.

In the case of ω_2 equal to one, the pixel p belongs to the uncertainty zone; hence, a different filtering strategy has to be used to reduce the noise in the object boundaries while avoiding the blurring effect. The proposed spatial term $f_2(\bullet)$ is a median filter applied only to those pixels in the neighborhood Ω^p that are similar to the pixel p in the color space according to the range term $g_2(\bullet)$

$$f_2(\bullet) = med \left\{ \widehat{D}^q \epsilon \Omega^p / g_2(I^p, I^q_m) = e^{-\frac{1}{2}\left(\frac{I^p - I^q_m}{\sigma_I}\right)^2} \geq th_{color} \right\}. \tag{5}$$

The range term follows a Gaussian profile, and operates in the Lab color space. The color information helps to discriminate the pixels in Ω^p that belong to different objects, thus, preserving the correct depth boundaries. In fact, we are considering that locally each object is characterized by pixels of similar color and similar depth. It is worth noting that, in case of foreground pixels, the range term is calculated by analyzing the actual depth map $\widehat{D}$ instead of $\widehat{D}_m$, U identifies the regions near depth discontinuities of D and I is used instead of I_m (there are no depth and color model values for *Fg* pixels).

The filtered depth map $\widehat{D}_f$ is then processed by the hole filling block to replace the *nmd* pixels (if they are still present) with locally consistent depth values. Given a *nmd* pixel p, the joint-bilateral filter is applied to interpolate its value D^p_f

$$D^p_f = H(C_{map}, \Omega^p) / k_p \sum_{q \epsilon \Omega^p} \widehat{D}^q_f f_1(p, q) g_2(I^p_m, I^q_m) \tag{6}$$

where $H(C_{map}, \Omega^p)$ evaluates the reliability of the neighborhood Ω^p based on the corresponding values of C_{map}; $f_1(\bullet)$ is the spatial term [also used in (3)] and $g_2(\bullet)$ is the range term considering color information as defined in (5) (obviously, the depth similarity cannot be computed). The advantage of this approach is that the *nmd* pixels are substituted by a value obtained by filtering neighbor depth values weighted by their color similarity, thus, discriminating from pixels that belong to different objects. Also, in this case, we are assuming that locally each object is characterized by pixels of similar color and similar depth; hence, the higher the correlation (considering the color space) and the proximity of the pixels in Ω^p, the higher their contribution to the *nmd* pixel filtering. Moreover, only those *nmd* pixels that guarantee a reliable neighborhood for their depth-value interpolation are computed. The expression of the function $H(C_{map}, \Omega^p)$ is

$$H(C_{map}, \Omega^p) = \begin{cases} 1 & \text{if } \frac{count_{q \epsilon \Omega^p}\left[C^q_{map} > c_{th}\right]}{Area(\Omega^p)} \geq th_{\%} \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

where $H(C_{map}, \Omega^p)$ is a binary function that evaluates which *nmd* pixels have to be filtered; the basic idea behind the introduction of this term in (6) is that only those *nmd* pixels that have a neighborhood Ω^p with a sufficient number of reliable depth measurements can be substituted with the new depth value D^p_f. For a given neighborhood Ω^p, all the q pixels in the C_{map} that have a value greater than the threshold c_{th} are included in the filtering process. The fraction of the pixels in Ω^p that meet the previous condition is evaluated, and if it is greater than the threshold $th_{\%}$, the *nmd* pixel at position p is filtered as shown in 6 with $H(C_{map}, \Omega^p) = 1$.

It is worth noting that (6) assumes that the *nmd* pixel at position p is part of the background. If the *nmd* pixel belongs to the foreground, (6) is applied with $H(\bullet)$ considering C^q_{map}

equal to 1 for all the foreground pixels with a depth value, and $g_2(\bullet)$ uses the color image I.

C. Adaptive Model Update: AMU

The proposed strategy is based on the estimation of two independent models of the quasi-static elements of the scene (background): a color-based model I_m and a depth-based model D_m. These models need to be iteratively updated to detect and, eventually, include modifications in the background (e.g., changes in illumination or new objects that become part of the background). Moreover, the iterative update allows reducing temporal fluctuations of depth data, hence obtaining a more reliable depth, and color, model.

The most reliable depth values of D_m are included in the filtering process (3) to guarantee an accurate selection of the filter weights, and in the hole filling strategy to interpolate new depth values for *nmd* pixels with consistent data by considering C_{map}. The color information contained in the model I_m determines the filter weights for the pixels included in the U map, and for the *nmd* pixels.

In our approach, the MoG background modeling algorithm [18] is proposed for both models. Its main features are very attractive for our application: accurate estimation of quasi-static backgrounds, adapting to new background configurations and gradual changes, and it models each pixel independently. The latter feature is very important since the proposed depth-color fusion strategy is based on the local improvement of the depth map; hence, an adaptive pixel-wise background modeling is well suited to accomplish this goal. Moreover, for the depth-based model we adapt, for each pixel, its parameters to the level of noise as we propose for the AF module. In the MoG model, the probability to find a pixel at time t of intensity X is defined as a mixture of Gaussians

$$P(X_t) = \sum_{i=1}^{K} \omega_{i,t}\eta(X_t, \mu_{i,t}, \Sigma_{i,t}) \tag{8}$$

where K is the number of Gaussians, $\omega_{i,t}$ is the weight associated to the i_{th} Gaussian η at the time t with mean $\mu_{i,t}$ and covariance matrix $\Sigma_{i,t}$. For the depth-based model, the Gaussians have a single dimension, and the model is updated using the filtered depth map D_f. In the case of the color-based model I_m, Gaussians have three components (Lab color space is proposed in this paper) that are in general assumed independent and the covariance matrix can be computed as $\Sigma_{i,t} = \sigma_{i,t} I$. The weight of a Gaussian measures the accuracy with which it models the value of the corresponding pixel.

The MoG algorithm is composed by two main phases: the estimation of whether or not a pixel belongs to the background model and the parameters update. The statistical test to classify each incoming pixel as belonging to the background and foreground is performed in the CDObj module (Section III-A). The multimodal background model is considered to be formed by the distributions that have a high ratio between their weight and standard deviation ($r_{i,t} = \omega_{i,t}/\sigma_{i,t}$): a high value for $r_{i,t}$ means that the i_{th} distribution has modeled very well the pixel in the past ($w_{i,t}$ is high) and that its value has a low variability (low $\sigma_{i,t}$) and is close to the mean ($\mu_{i,t}$) of the Gaussian. The first B distributions that exceed a certain threshold T are used for the background model

$$B = argmin_b \left(\sum_{i=1}^{b} \omega_{i,t} \geq T \right). \tag{9}$$

T is a measure of the minimum portion of the data that should be accounted for by the background. For small values of T, a background modeled by few distributions is obtained; the limit is a single Gaussian distribution. If T is high, a multimodal background model is obtained. A pixel belongs to one of the K distributions if (10) is satisfied

$$\sqrt{\left(X_{t+1} - \mu_{i,t}\right)' \Sigma_{i,t}^{-1} \left(X_{t+1} - \mu_{i,t}\right)} < 2.5. \tag{10}$$

Equations (9) and (10) are used for both depth-based and color-based models, although for the depth model the ranking parameter r has to be modified. In fact, the standard deviation of the depth measurements $\sigma_{i,t}$ is related to the measured depth value. Therefore, pixels corresponding to object points close to the camera show always smaller $\sigma_{i,t}$ values than those for points located further; thus, introducing a possible bias in the ranking procedure. To limit this bias, we propose normalizing $r_{i,t}$ with a parameter σ_{min} that is selected according to the quadratic law that relates distance $\mu_{i,t}$ and noise dispersion (Fig. 2). This parameter represents the minimum allowed standard deviation for a Gaussian distribution modeling the depth background model.

If the pixel belongs to one of the background distributions, it is classified as a background pixel; otherwise it is classified as a foreground pixel. If a match [according to (10)] is found, the parameters of the matching Gaussian, in the case of 1-D distribution, are updated as follows:

$$\begin{aligned}
&\omega_{i,t+1} = \omega_{i,t}(1-\alpha) + \alpha * M \\
&\rho = \alpha\eta(X_t, \mu_{i,t}, \Sigma_{i,t}) \\
&\mu_{i,t+1} = \mu_{i,t}(1-\rho) + \rho X_t \\
&\sigma_{i,t+1}^2 = \sigma_{i,t}^2(1-\rho) + \rho\left(X_{t+1} - \mu_{i,t+1}\right)^2
\end{aligned} \tag{11}$$

where α, called learning rate, determines the adaptation to changes in the scene and the speed of the incorporation of foreground objects to the background. The learning rate indicates the influence that the last data have on the Gaussian distribution parameters. For the unmatched Gaussians, all the parameters remain unchanged except the weight that is updated with $M = 0$ in (11).

It is worth noting that very stable measurements can lead to very small values of Gaussian variances, thus, limiting the adaptability of the model to gradual and small measurements variations. In general, to avoid this problem the estimated standard deviation is limited by a minimum value. The selection of this parameter is straightforward (and fixed for every pixel) if the MoG is used in a color space. The choice of this value is more critical in the case of the depth-based model due to the distance-dependent noise; hence, we use the parameter $\sigma_{min} = \sigma_{noise}$ (Fig. 2). In this way, also the depth model is dynamically adapted to the distance dependent noise.

Another challenging issue is the MoG initialization, since moving objects in the scene may make difficult to properly train the model. We tackle this problem by selecting the

learning rate as presented in [19], where α is initially set to $1/N$, where N is the number of processed frames, until an α_{min} value is reached. Therefore, during initialization the first frames have a great impact on the distributions parameters, thus, preventing from incorporating the moving objects to the background model. Furthermore, the filtering process starts to incorporate the Bg information (independently for each pixel) only when at least one Gaussian is reliably modeling a static scene element. In particular, the values of the weight and the standard deviation (with respect to the corresponding σ_{min}) of the distributions are checked. In this way, pixels in an area containing static objects will rapidly reach these conditions, hence, can be included in the filtering process as model of the static background; on the contrary, depth measurements of regions containing moving objects are not wrongly included in the static background.

It is worth noting that the two models are independent, and their combination is fundamental to refine the depth-based Fg/Bg detection. The color model is updated by considering only the color-based object detection.

IV. Results

In this section, we present the results obtained with the proposed color-depth fusion strategy applied to different datasets. Tests have been conducted to individually highlight the main contributions of our approach, to demonstrate its strengths and capabilities, and to compare its performance with other proposals in the state of the art. In Section IV-A, we investigate how the adaptive joint bilateral filter allows to efficiently reduce the distance dependent spatial noise. In Section IV-B, we evaluate the advantages of our solution to the problem of the noisy depth measurements at sharp depth transitions, demonstrating how the color information is fundamental to refine depth measurements at object boundaries. The performance of the proposed hole filling strategy is addressed in Section IV-C, providing both quantitative and qualitative results. Finally, indoor detection of moving objects and modeling improvement capabilities are explored in Section IV-D.

A. Adaptive Filtering

In this section, it is shown how the proposed approach efficiently reduces the distance-dependent spatial depth-map noise on continuous surfaces that contain small details not generating sharp depth transitions (the performance of the filtering approach at depth-map borders is further investigated in the next section). To assess the performance of the algorithm, a synthetic dataset is proposed to which distance dependent noise as shown in Fig. 2 has been added. Images in the dataset represent rectangular panels facing the camera at different distances, having each six small box-like cavities that simulate small details in a depth map. The details depth value d is selected as $d = d_f + 2 * \sigma_{noise}$, where d_f is the depth value of the panel and σ_{noise} is its corresponding distant-dependent noise level. To simulate the Kinect-acquisition pattern, Gaussian noise ($\mu = d_f, \sigma = \sigma_{noise}$) is added.

This dataset is used to compare the proposed AF strategy with its nonadaptive implementation, the guided filter (GF) proposed in [13], and the cascade of median and bilateral filters (MBF) proposed in [5]. Performance is assessed through the normalized absolute error (NAE) filtering gain, defined as: $1 - (NAE_{filtered}/NAE_{raw})$, where NAE_{raw} is the NAE value of the raw depth-map data, and $NAE_{filtered}$ is that obtained after filtering. Fig. 6 reports, for each proposed approach, the evolution of the NAE gain value (in %) with d_f. Let us compare the adaptive approach, in which σ_d in (4) is set to a value σ_{noise} that depends on the depth measurement value (following Fig. 2), with its nonadaptive implementation that use fixed values for σ_d. It is worth noting that the proposed adaptive approach guarantees an almost constant gain (above 70%) for all the depth range. On the contrary, the nonadaptive implementation provides comparable results only for a limited interval. In the case of using small values for the filter (σ_d = 5 mm, black dashed line), the performance decays abruptly with the distance: the bilateral filter is simply replicating the noisy measurements, and for distances greater than 2500 mm the gain plumbs down to values close to zero. In the case of intermediate (σ_d = 25 mm, blue dashed line) and large values (σ_d = 50 mm, red dashed line), the gain keeps fairly high values (always lower than those with the proposed adaptive approach), for distances below that for which the selected parameter value is optimum. This is due to the fact that considering σ_d greater than the noise level the filter has a strong low pass behavior on the depth information, thus blurring depth details. Their performance has a peak for those regions at the distance for which the parameter has been selected, fast decaying for larger ones as it happened for small values of σ_d. Regarding the comparison with the GF (red line) and MBF (blue line) methods, they provide quite stable NAE gain values, although the proposed strategy outperforms them.

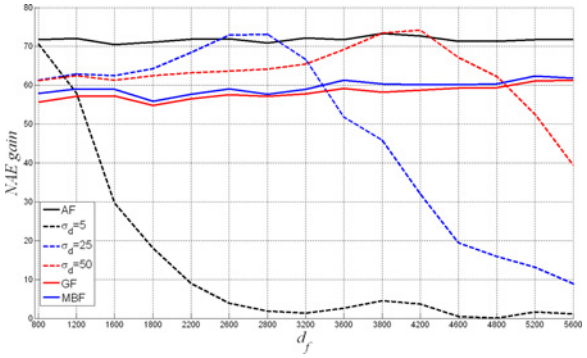

Fig. 6. NAE gain as a function of d_f: AF black line, GF red line, MBF blue line. Fixed filter parameters: σ_d = 5 dashed black line, σ_d = 25 dashed blue line and σ_d = 50 dashed red line.

The impact of using a fixed parameterization depends on the scene contents, on how the depth of its static and moving objects differ from that for which the selected σ_d is optimum. Therefore, the advantage of the AF is clear: it guarantees the best performance regardless of the depth values of the objects in the scene, while preserving depth map details.

B. Noisy Object Boundaries Filtering

In this section, we analyze the performance of the proposed strategy near object boundaries; in particular, we want to highlight the importance of our proposed combination of color and depth information to efficiently improve the depth map

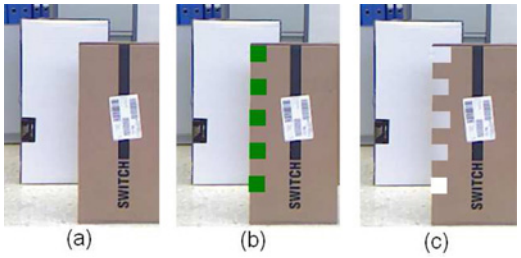

Fig. 7. Setup I: boundaries refinement tests under different color condition. (a) High color contrast, HC. (b) Green patches, GC. (c) White patches, WC.

near object borders. In these experiments, we use two different setups shown in Figs. 7 and 9.

In Setup I (Fig. 7), there are two overlapping boxes placed parallel to the camera at 146 cm (brown box) and 186 cm (white box). In this experiment, the impact of different color textured borders on the performance of the proposed strategy is evaluated. In particular, we propose four different color conditions for the boundaries between boxes. High color contrast (HC), shown in Fig. 7(a). No color contrast (NC), where the lack of color discontinuity (or a poorly illuminated scene) is simulated disabling the use of color data by the proposed strategy, leading to an empty U map. The last two cases consider different color patches to test the proposed approach containing misleading colors close to depth discontinuities: the green (GC) and white (WC) cases shown in Fig. 7(b) and (c).

The performance of the proposed algorithm has been compared with that of different state-of-the-art algorithms: the GF proposed in [13]; the bilateral filter (BLC) proposed in [10] and the cascade of MBF proposed in [5]. Performance is assessed through the NAE filtering gain. Table I reports the mean and standard deviation of the NAE filtering gain (in percent), computed on sequences of 40 frames, for the proposed strategy with the four color conditions and for the three above-mentioned algorithms.

Let us first analyze the performance of the proposed algorithm with different color conditions: as expected, the boundaries of the NAE gain are defined by the HC (high contrast, full support of color information to the depth map filtering), and the NC case (no color information available). In the case of textured color borders, although the gain values decrease, particularly when misleading colors are present as in the WC case, the color-depth combination guarantees always higher gain values with respect to the NC case. Therefore, color and depth data combination is fundamental to efficiently refine object borders also in the cases in which color similarity and depth similarity do not completely match. Regarding the comparison with the other algorithms, the proposed one renders always higher gain values with lower dispersion. In particular, it outperforms the GF and MBF depth-based approaches, even in its only-depth-data configuration (NC case). Compared with BLC strategy, that also takes into account the color information, the proposed strategy results in double gain values with half dispersion: the use of Gaussian kernel in the BLC bilateral filter introduces a blurring effect in the borders that severely downgrade its performance.

TABLE I
OBJECT BOUNDARIES FILTERING: NAE GAIN (%)

Algorithm	Mean	Std. Dev.
Proposed (HC)	62.5	4.8
Proposed (GC)	61.9	4.3
Proposed (WC)	43.4	6.5
Proposed (NC)	21.5	8.9
BLC [10]	30.3	8.3
MBF [5]	19	9.7
GF [13]	8.8	10.2

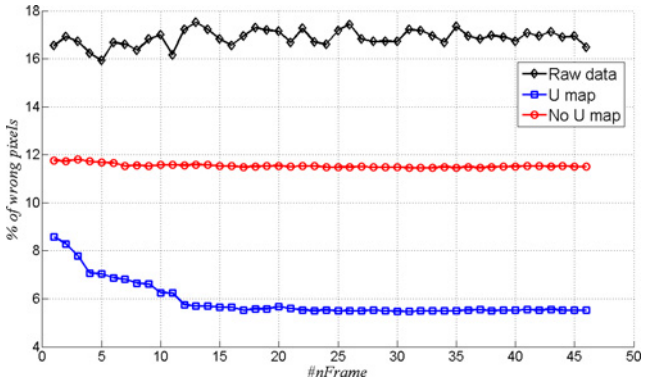

Fig. 8. Percentage of wrong pixels in the object borders. Unfiltered data (black), proposed strategy without U map (red), and with U map (blue).

Setup II [Fig. 9(d)] is composed by a file cabinet (*fc*) and a box (*box*), whose front sides are at 222 cm and 265 cm from the camera, both leaning against a wall at 292 cm from the camera. We compare the performance of our approach with and without considering the U map. Filtered depth maps are compared with a hand-labeled ground-truth of the *fc*-wall and box-wall boundaries regions. The performance is measured as the percentage of wrong pixels: pixels whose values do not match the ground truth ones.

The curves in Fig. 8 demonstrate that the U map highly improves the filtering results: the stabilization of the background model along time smoothly decrease the percentage of wrong pixels until a minimum is reached. Not considering the U map still results in stabilized borders, although they do not fit the actual object ones. Moreover, it has a different impact in the filtering process of the object borders depending on the between objects depth gap. In the case of *fc*, as the *fc-wall* depth gap is higher than the level of noise in the depth measurements, the adaptive bilateral filter preserves the irregular borders in depth without blurring them. On the contrary, in the case of box, as the box-wall depth gap is small (smaller than the depth measurements noise at those distances) the bilateral filter introduces a blurring effect in the depth borders. A detail of the resulting depth maps obtained with and without the U map is shown in Fig 9(b–c). These tests demonstrate that the combination of depth and color information is fundamental to accurately refine object boundaries; in fact, the proposed approach over performs other techniques presented in the literature.

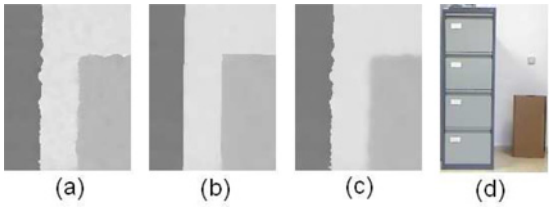

Fig. 9. Setup II. (a) Raw depth map detail. (b) Proposed strategy considering the U map. (c) Proposed strategy without U map. (d) Setup color image.

TABLE II
HOLE FILLING COMPARISON

Algorithm	NAE_{nmd}	NAE_{zone}
Proposed	0.8308	0.0601
INP [15]	0.8380	0.0854
RMF [14]	0.8619	0.1097
MBF [5]	0.8614	0.1075
GF [13]	0.8648	0.1117

C. Hole Filling

The AF module accurately interpolates the *nmd* pixels in the depth map: a local combination of reliable depth information weighted by color similarity is proposed. Its performance is compared with four proposals: the inpainting algorithm proposed in [15] (INP), the recursive median filter used in [14] (RMF), the cascade of MBF proposed in [5], and the normalized convolution combined with the GF proposed in [13]. An indoor scene recorded in our laboratory, similar to the one proposed in [15], is used. The dataset used for these experiments is shown in Fig. 10(a), where a background wall and a box stand parallel to the image plane at 3270 mm and 1850 mm from the camera.

Table II contains the mean value of the NAE obtained on a sequence of 40 frames. The second column reports the NAE calculated only on the *nmd* pixels, and the third one reports the NAE calculated considering also the regions surrounding them.

As it can be noticed, the proposed method guarantees the minimum value of both NAE_{nmd} and NAE_{zone}, while resulting in an excellent adaptation to the object actual boundaries as shown in Fig. 10(b). The closest NAE_{nmd} results are obtained with the inpainting strategy INP, as it considers similar factors: neighborhood depth data, spatial closeness, and color similarity. However, its performance significantly deteriorates when neighboring areas are considered (NAE_{zone}): the use of temporal information in our model helps stabilizing and refining the regions close to the *nmd* pixels, thus, improving their interpolation and the overall quality of the depth map. Moreover, the interpolated boundaries are noisy and do not fit the object boundaries as shown in Fig. 10(c). Regarding the algorithms that do not consider color information in the interpolation process (RMF, MBF, and GF), worse results are obtained both, in terms of NAE values and in the resulting quality of the filtered depth maps as shown in Fig. 10(d)–(f).

Results obtained with the different strategies using the object database proposed in [14] fully support the previous

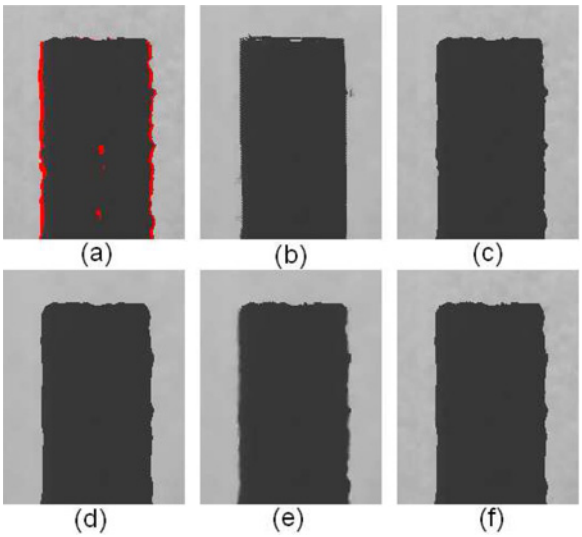

Fig. 10. Hole filling results. (a) Depth-map raw data. (b) Proposed strategy. (c) INP. (d) MBF. (e) GF. (f) RMF.

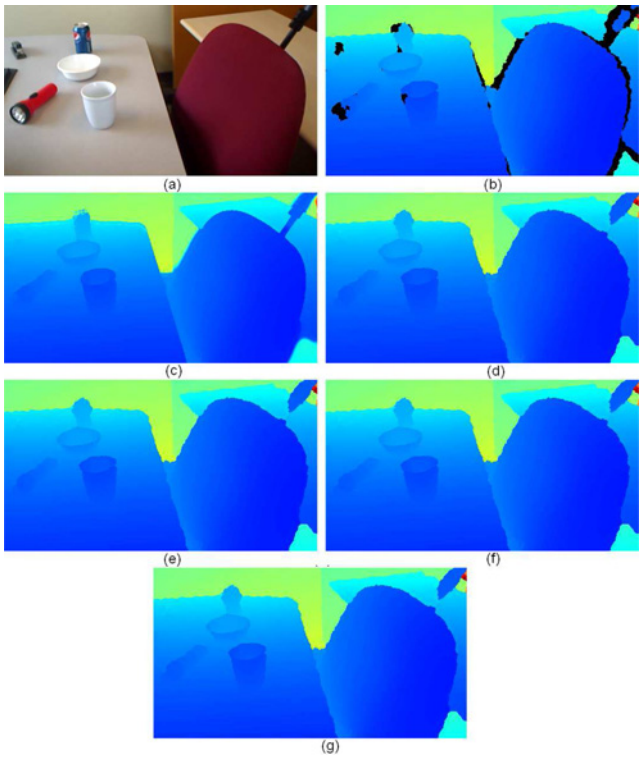

Fig. 11. Details of frame 8 meeting sequence of dataset [14]. (a) Color data. (b) Depth data. (c) Proposed strategy. (d) INP. (e) MBF. (f) GF. (g) RMF.

analysis. An example is presented in Fig. 11, where depth maps are reported in a different color palette to highlight objects details. The raw depth map in Fig. 11(b) has the *nmd* pixels regions marked in black. As it can be observed in Fig. 11(c), the proposed method improves significantly the accuracy of the depth maps region containing *nmd* pixels: boundaries are completed and refined, leading to locally smooth regions with accurate depth values. On the contrary, the results obtained with the other methods not only lead to

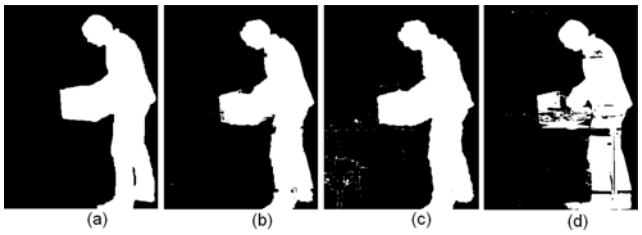

Fig. 12. *Fg* regions of frame 141. (a) Ground truth. (b) Proposed method. (c) MoG_D. (d) MoG_C.

TABLE III
DETECTION ACCURACY

Algorithm	TE	FN	FP	S
MoG_C	2.95	23.45	0.35	0.67
MoG_D	1.37	2.71	1.21	0.83
Proposed: $D_m + I_m$	0.95	7.20	0.16	0.87
Binary [22]	1.49	1.82	1.46	0.82
MoG_{C+D} [21]	1.76	1.74	2.77	0.8

less refined and coarser object boundaries (e.g., the chair), but to erroneous interpolations (e.g., ark handler behind the chair).

D. Dynamic Indoor Scene Modeling

We present the results of the proposed strategy applied to an indoor scene, captured in our labs, containing both static and moving objects; in particular, we demonstrate how our proposal guarantees modeling of the static background and at the same time an accurate refinement of the moving objects silhouette. For performance quantitative evaluation, a hand-labeled ground truth sequence of 150 frames is provided.

For the best of authors' knowledge, there are only few works in literature that combine depth and color information for foreground/background segmentation (see the recent review in [20]). In [21], an extension of the MoG model has been proposed to fuse color and depth data from a stereo camera. Each background pixel is modeled as a mixture of 4-D Gaussian distributions that consider the color components and depth. Color and depth are considered independent and equations similar to the ones presented in Section III-C are used to update the distribution parameters. In [22], the depth obtained by a time-of-flight camera is combined with color information: the depth and color-based foreground masks are combined with an OR logical operation.

We evaluate the accuracy of the proposed strategy with the above-mentioned methods (we refer to them as MoG_{C+D} and Binary) and with the results obtained applying a MoG strategy only to the depth data (MoG_D) or to the color data (MoG_C). Algorithms performance is evaluated with four indicators: false positive (FP), that is, the fraction of the *Bg* pixels that are marked as *Fg*; false negative (FN), that is, the fraction of *Fg* pixels that are marked as *Bg*; the total error TE, that is, the fraction of misclassified pixels, and the similarity measure S proposed in [23]: a nonlinear measure that merges FP and FN and it is close to 1 if detected *Fg* regions correspond to the real ones, otherwise its value is close to 0.

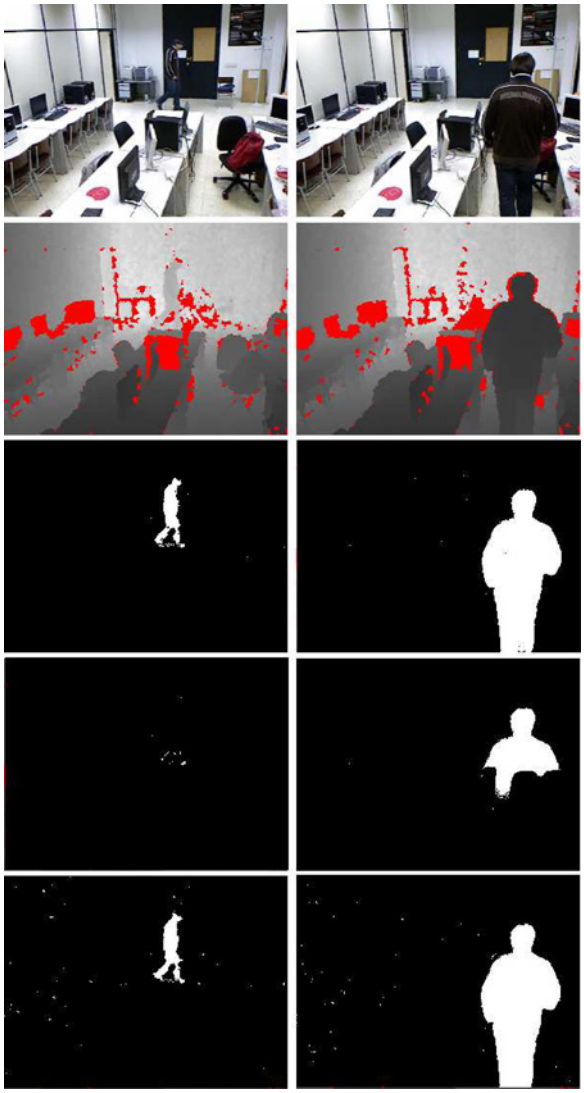

Fig. 13. Detection accuracy with different model parameters. Color images (first row) and corresponding raw depth data (second row). Detected *Fg* with: proposed adaptive σ_{min} (third row); fixed value of σ_{min}=45 mm (fourth row); and σ_{min}=5 mm (fifth row).

Table III reports the results obtained by the different approaches. The use of color data only (MoG_C), leads to a very high FN value due to the well-known problem of color camouflage. The ratio of FN is reduced by MoG_D since the depth data provide very compact object silhouettes, although the FP value is increased mainly due to the noisy boundaries. The proposed approach allows reducing simultaneously the value of FN and FP, while rendering the highest value for *S*. It outperforms the proposals in [21] and [22], providing higher accuracy results: lowest TE and highest *S*.

Fig. 12 reports the foreground masks obtained with the proposed method (b), MoG_D (c), MoG_C (d); the ground truth is shown in (a). As it can be noticed, the proposed method efficiently combines the compact silhouette of MoG_D and the accuracy at object contours obtained with MoG_C, eliminating also several false positive detections obtained by MoG_D.

Another important feature of the proposed background modeling strategy is its noise-adaptive parameterization as

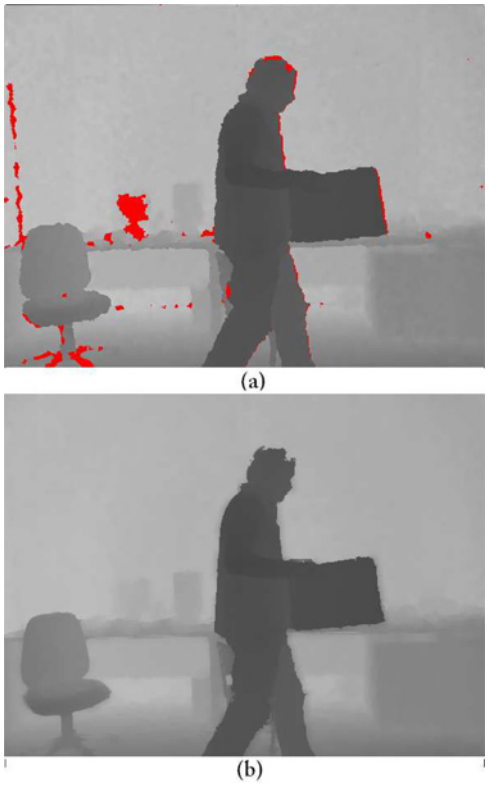

Fig. 14. Indoor environment. (a) Raw depth data. (b) Filtered depth map.

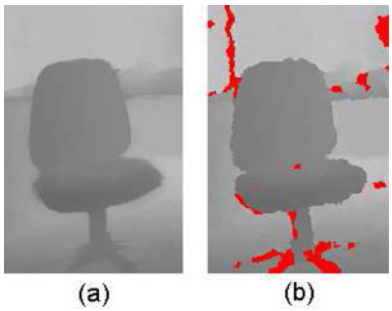

Fig. 15. Static object detail. (a) Filtered depth map. (b) Raw depth map.

explained in Sections III-B and III-C. Fig. 13 reports two depth maps of an indoor scene with a moving object. The use of adaptive-parameters in the models (third row in the figure) allows rendering accurate *Fg* regions simultaneously for situations in which the moving object is either far or close to the camera and, at the same time, reducing the depth-map measurements high level of noise on very far located objects (i.e., the wall). On the contrary, using a fixed σ_{min}, it is difficult to obtain simultaneously an acceptable segmentation for both, far and close to the camera moving objects. In fact, by using a large value for σ_{min} to reduce the temporal fluctuation at far distance, Fig.13 (fourth row), part of the *Fg* regions are incorporated to the background. If a small value of σ_{min} is used to improve the *Fg* detection accuracy, Fig. 13 (fifth row), the false detections due to the noise at far distances increases.

Fig. 14(a) reports raw depth data provided by the Kinect, which is severely affected by the noise problems described in the introduction. The corresponding point cloud model is

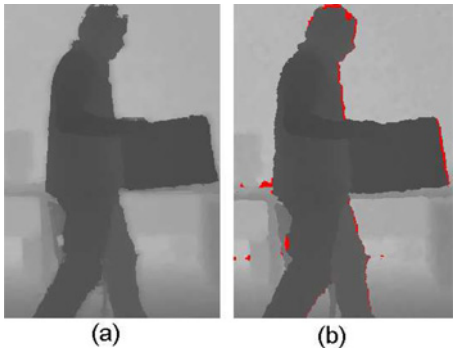

Fig. 16. Moving object detail. (a) Filtered depth map. (b) Raw depth map.

Fig. 17. Point cloud generated from the raw depth map.

Fig. 18. Point cloud generated from the refined depth map.

presented in Fig. 17, where the resulting low quality of the 3-D model can be visually evaluated. The proposed strategy improves dramatically the accuracy of the depth map as shown in Fig. 14(b): in the background areas, depth measurements are stabilized, showing refined and more accurate object boundaries, and locally smoothed depth regions (Fig. 15). Definition of the foreground regions, corresponding to a moving person carrying a box (see details in Fig. 16), has very significantly improved: object boundaries fit much better the actual moving object silhouette. Fig. 18 renders the corresponding point cloud where artifacts and errors have been reduced: models are coherently completed, such as the pc monitor; both static and moving elements of the scene are sharply defined; see, for example, the more homogeneous flat regions in the wall and the file cabinet under the table.

V. Conclusion

Kinect depth maps are characterized by different noise related problems that have to be efficiently tackled in order to improve their accuracy and to broaden their application possibilities. In this paper, we introduce a depth-color fusion strategy for 3-D modeling of indoor scenes with the Kinect device. The background elements of the scene are accurately modeled by using independently depth and color data; the obtained models are used to detect moving elements in the scene. The acquired depth data is processed with an innovative adaptive filter that took into account depth data, color information, a depth-based edge-uncertainty map, and the detected foreground regions. The main innovative features of the proposed strategy are the adaptive filtering process based on the introduction of the edge-uncertainty map; the combination of the adaptive depth model and the color for the object detection. Another attractive feature of the system is its pixel-wise processing approach that guarantees a feasible parallel implementation with GPU architecture for real-time application. Quantitative and qualitative evaluation results present that the proposed strategy efficiently improves the accuracy of Kinect depth maps.

References

[1] Microsoft Corporation. *Kinect for Xbox 360* [Online]. Available: http://www.xbox.com/en-US/kinect/default.htm

[2] X. Suau, J. Ruiz-Hidalgo, and J. R. Casas, "Real-time head and hand tracking based on 2.5-D data," *IEEE Trans. Multimedia*, vol. 14, no. 3, pp. 575–585, May 2012.

[3] J. Shotton, R. Girshick, A. Fitzgibbon, T. Sharp, M. Cook, M. Finocchio, R. Moore, P. Kohli, A. Criminisi, A. Kipman, and A. Blake, "Efficient human pose estimation from single depth images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PP, no. 99, p. 1, Nov. 2012.

[4] *OpenNI Orgaization. OpenNI User Guide* [Online]. Available: http://www.openni.org/

[5] A. Maimone, J. Bidwell, K. Peng, and H. Fuchs, "Enhanced personal autostereoscopic telepresence system using commodity depth cameras," *Comput. Graphics*, vol. 36, no. 7, pp. 791–807, 2012.

[6] L. Chen, H. Wei, and J. Ferryman, "A survey of human motion analysis using depth imagery," *Pattern Recognit. Lett.*, 2013.

[7] D. Wu, F. Zhu, and L. Shao, "One shot learning gesture recognition from RGBD images," in *Proc. IEEE Comput. Soc. Conf. Comput. Vision Pattern Recognit. Workshops*, Oct. 2012, pp. 7–12.

[8] K. Khoshelham and S. O. Elberink, "Accuracy and resolution of kinect depth data for indoor mapping applications," *Sensors*, vol. 12, no. 2, pp. 1437–1454, 2012.

[9] F. Menna, F. Remondino, R. Battisti, and E. Nocerino, "Geometric investigation of a gaming active device," in *Proc. SPIE*, vol. 8085, 2011, p. 80850G.

[10] M. Camplani, T. Mantecon, and L. Salgado, "Accurate depth-color scene modeling for 3-D contents generation with low cost depth cameras," in *Proc. 19th IEEE Int. Conf. Image Process.*, Sep.–Oct. 2012, pp. 1741–1744.

[11] M. Camplani and L. Salgado, "Efficient spatio-temporal hole filling strategy for Kinect depth maps," in *Proc. Three-Dimensional Image Process. Applicat. II*, vol. 8290, no. 1. 2012, p. 82900E.

[12] S. Matyunin, D. Vatolin, Y. Berdnikov, and M. Smirnov, "Temporal filtering for depth maps generated by Kinect depth camera," in *Proc. 3DTV Conf.: True Vision Capture Transmission Display 3-D Video*, May 2011, pp. 1–4.

[13] J. Wasza, S. Bauer, and J. Hornegger, "Real-time preprocessing for dense 3-D range imaging on the GPU: Defect interpolation, bilateral temporal averaging and guided filtering," in *Proc. IEEE Int. Conf. Comput. Vision Workshops*, Nov. 2011, pp. 1221–1227.

[14] K. Lai, L. Bo, and X. Ren, "A large-scale hierarchical multi-view RGBD object dataset," in *Proc. Int. Conf. Robot. Autom.*, 2011, pp. 1817–1824.

[15] F. Qi, J. Han, P. Wang, G. Shi, and F. Li, "Structure guided fusion for depth map inpainting," *Pattern Recognit. Lett.*, vol. 34, no. 1, pp. 70–76, Jun. 2013.

[16] A. Telea, "An image inpainting technique based on the fast marching method," *J. Graphics Tools*, vol. 9, no. 1, pp. 23–34, 2004.

[17] G. Petschnigg, R. Szeliski, M. Agrawala, M. Cohen, H. Hoppe, and K. Toyama, "Digital photography with flash and no-flash image pairs," *ACM Trans. Graphics*, vol. 23, no. 3, pp. 664–672, 2004.

[18] C. Stauffer and W. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Aug. 1999, pp. 246–252.

[19] P. Kaewtrakulpong and R. Bowden, "An improved adaptive background mixture model for real-time tracking with shadow detection," in *Proc. 2nd Eur. Workshop Adv. Video-Based Surveillance Syst.*, 2002, pp. 135–144.

[20] M. Cristani, M. Farenzena, and D. Bloisi, and V. Murino, "Background subtraction for automated multisensor surveillance: A comprehensive review," *EURASIP J. Adv. Signal Process.*, article no. 43, pp. 1–24, 2010.

[21] G. Gordon, T. Darrell, M. Harville, and J. Woodfill, "Background estimation and removal based on range and color," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, vol. 2. Jun. 1999, pp. 2459–2464.

[22] J. Leens, O. Barnich, S. Piérard, M. Droogenbroeck, and J.-M. Wagner, "Combining color, depth, and motion for video segmentation," in *Computer Vision Systems*, vol. 5815. Berlin, Germany: Springer, 2009, pp. 104–113.

[23] L. Li, W. Huang, I. Y.-H. Gu, and Q. Tian, "Statistical modeling of complex backgrounds for foreground object detection," *IEEE Trans. Image Process.*, vol. 13, no. 11, pp. 1459–1472, Nov. 2004.

Massimo Camplani (M'10) received the M.S degree (Hons.) in electronic engineering in 2006, and the Ph.D. degree in electronic and computer engineering in 2010, from the Università degli Studi di Cagliari, Italy.

Since 2010, he has been a member of the Grupo de Tratamiento de Imágenes (Image Processing Group) of the Universidad Politécnica de Madrid, Madrid, Spain. His current research interests include the areas of computer vision.

Dr. Camplani was a recipient of the Marie Curie-COFUND Grant in April 2011.

Tomás Mantecón received the Telecommunication Engineering degree (five-year engineering program) in 2012 from the Universidad Politécnica de Madrid (UPM), Madrid, Spain. He is currently pursuing the Ph.D. degree at the same University.

Since 2013, he has been a member of the Grupo de Tratamiento de Imágenes (Image Processing Group) of the UPM. His current research interests include the areas of video analysis and processing.

Luis Salgado (M'05) received the Telecommunication Engineering degree in 1990 and the Ph.D. degree in communications with summa cum laude in 1998, both from the E.T.S.I. Telecomunicación, Universidad Politécnica de Madrid (UPM), Spain.

Since 1990, he has been a member of the Image Processing Group (GTI) of the UPM. Since 1996, he has been a member of the faculty of the UPM, formerly as a Teaching Assistant, and currently as an Associate Professor (tenure in 2001) of Signal Theory and Communications in the Department of Signals, Systems, and Communications. From September 2012, he has also joined the Universidad Autónoma de Madrid, Madrid, Spain, as an Associate professor at the Escuela Politécnica Superior (VPULab Group, Department of Electronics and Communications Technology). His current research interests include video analysis, processing and coding.

Dr. Salgado is an Associate Editor of the *Journal of Real-Time Image Processing*. He has been a member of the Scientific and Program Committees of several international conferences and has been the auditor and evaluator of European research programs since 2002. He has participated in many national and international research projects.