Multi-modal User Identification and Object Recognition Surveillance System

Albert Clapés

Dept. Matemàtica Aplicada i Anàlisi, Facultat de Matemàtiques, Universitat de Barcelona, Gran Via de les Corts Catalanes 585, 08007, Barcelona, Spain.
aclapes@gmail.com

Abstract

We propose an automatic surveillance system for user identification and object recognition based on multi-modal RGB-Depth data analysis. We model a RGBD environment learning a pixel-based background Gaussian distribution. Then, user and object candidate regions are detected and recognized using robust statistical approaches. The system robustly recognizes users and updates the system in an online way, identifying and detecting new actors in the scene. Moreover, segmented objects are described, matched, recognized, and updated online using view-point 3D descriptions, being robust to partial occlusions and local 3D viewpoint rotations. Finally, the system saves the historic of user-object assignments, being specially useful for surveillance scenarios. The system has been evaluated on a novel data set containing different indoor/outdoor scenarios, objects, and users, showing accurate recognition and better performance than standard state-of-the-art approaches.

Keywords: Multi-modal RGB-Depth data analysis, User identification, Object Recognition, Intelligent Surveillance, Visual features, Statistical learning.

1. Introduction

Technology has reached a stage where mounting cameras to capture video imagery is cheap, but finding available human resources to sit and watch that imagery is expensive. Organizations often spend millions of dollars on video surveillance infrastructure consisting of hundreds or thousands of cameras. These camera feeds are usually backhauled to a central monitoring location

where some of them are recorded for a period of time on local video storage media, and some of them are displayed in real-time to one or more security personnel on a bank of video monitors. No matter how highly trained or how dedicated a human observer is, it is impossible to provide full attention to more than one or two things at the same time; and even then, only for a few minutes at a time. A vast majority of surveillance video is permanently lost without any useful intelligence being gained from it. Thus, it is essential to automatically collect and disseminate real-time information from the battlefield to improve the situational awareness of the users in a given scenario.

Several automatic approaches related to this topic has been published [1, 2, 3, 4, 5, 6]. These works base on Computer Vision techniques to examine the video streams to determine activities, events or behaviors that might be considered suspicious and provide an appropriate response when such actions occur. The detection of motion in many current tracking systems relies on the technique of background subtraction. By comparing incoming image frames to a reference image, regions of the image which have changed are efficiently located. There exists several works for background model representation and adaptation [7, 8, 2, 9, 10, 5]. However, there is a problem which has received little attention is model initialization. Often the assumption is made that an initial model can be obtained by using a short training sequence in which no foreground objects are present. However, in some situations, e.g., public areas, it is difficult or impossible to control the area being monitored. In such cases it may be necessary to train the model using a sequence which contains foreground objects. The ability to represent multiple modes for the background values allows some techniques to model motion which is part of the background [7, 10, 5]. The methods may also be grouped into those which estimate background values using temporal smoothing [7, 9, 10, 5, 3], and those which choose a single value from the set of past observations [11, 12]. In this sense, Mittal and Paragios [13] presented a motion-based background subtraction by using adaptive kernel density estimation. In their method, optical flow is computed and utilized as a feature in a higher dimensional space. They successfully handled the complex background, but the computation cost is relatively high. Some hybrid change detectors have been developed which combine temporal difference imaging and adaptive background estimation to detect regions of change [14, 15]. Huwer et al. [15] proposed a method of combining a temporal difference method with an adaptive background model subtraction scheme to deal with lighting changes. However,

none of these methods can adapt to quick image variations such as a light turning on or off. Li et al. [16] proposed a Bayesian framework that incorporates spectral, spatial, and temporal features to characterize the background appearance at each pixel. Their method can handle both the static and dynamic backgrounds and good performance was obtained on image sequences containing targets of interest in a variety of environments. The computation cost is relatively expensive for real-time video surveillance systems because of the computation of optical flow.

Most of vision-based automatic systems for surveillance analysis use to strategically locate different cameras in an environment for monitoring. When the surveillance purpose only requires to detect the presence of subjects in restricted areas, movement sensors are robust and simple to install. In those cases where one needs to identify allowed users or objects, as well as the membership of objects to different subjects, Computer Vision techniques (CV) use to be applied. These CV techniques have been studied for decades, and although huge improvements have been performed, still it is difficult to robustly identify users in visual data. Some common problems are: the wide range of human pose configurations, influence of background, illumination changes, partial occlusions, or different points of view, just to mention a few. In the case of object recognition the problem is even harder, since one has to deal with the same problematic than user detection but in smaller image regions. Some works have addressed the problem of developing complete vision systems for both object recognition and tracking in order to obtain a rough scene understanding [17, 18, 19]. However, recognition and tracking tasks are not integrated in a common spatio-temporal domain, and thus, occlusions and noise can generate false object appearance in the scene. Moreover, the tracking and the recognition are based on different kind of features so that the computational complexity of the whole system becomes very high. There exists other video surveillance methodologies focused on the analysis of anomalous behaviors or unusual objects. Previous approaches to recognition of suspicious behaviors or activities can broadly be classified into two classes of approaches: rule-based methods [20] and statistical methods without predefined rules [21, 22]. The statistical methods are more appealing, since they do not assume a predefined set of rules for all valid configurations. Instead, they try to automatically learn the notion of regularity from the data, and thus infer about the suspicious. Nevertheless, the representations employed in previous methods have been either very restrictive (e.g., trajectories of moving objects [21]), or else too global (e.g., a single small descriptor vector

for an entire frame [22]).

With the objective of improving the discriminability of relevant surveillance events in the scenes, some authors use calibrated cameras which are synchronized in order to obtain an approximation of the 3D representation of the scene. Although this approach can be useful in some situations, it requires from a perfect multi-camera synchronization, and a strategic location of each camera that could not be feasible in most real environments. Recently, with the appearance of the Depth maps introduced by the Kinect Microsoft device, a new source of information has emerged, and although its first applications have been devoted to video games, its potential can be exploited in several real applications, including automatic surveillance. With the use of depth maps, 3D information of the scene from a particular point of view is easily computed, and thus, working with consecutive frames, we obtain RGBDT information, from Red, Green, Blue, Depth, and Time data, respectively. This motivates the use of multi-modal data fusion strategies which can exploit the discriminatingly of the new data representation and increase generalization of classical CV and Pattern Recognition approaches.

Following the high popularity of Kinect and its depth capturing abilities, there exists a research interest for improving the current methods for human detection, tracking and scene interaction. With the arrival of Microsoft's Kinect, such sensing has suddenly reached wide consumer-level accessibility. Many researchers have obtained their first results in the field of human motion capture using this technology. In particular, Girshick and Shotton et al. [23, 24] present one of the greatest advances in the extraction of the human body pose from depth images, that also forms the core of the Kinect human recognition framework. Other recent work uses the skeletal model in conjunction with computer vision techniques to detect complex poses in situations where there are many interacting actors [25]. Through this technology are emerging work on reconstruction of dense surfaces [26], and 3D object detection [27, 28]. Much of this work is motivated by realtime applications.

Currently exists a steady stream of updates and tools that provide robustness and applicability to the device. In December 2010, OpenNI [29] and PrimeSense [30] released their own Kinect open source drivers and motion tracking middleware (called NITE [31]) for PCs running Windows (7, Vista and XP), Ubuntu and MacOSX. FAAST (Flexible Action and Articulated Skeleton Toolkit [32]) is a middleware developed at the University of Southern California (USC) Institute for Creative Technologies that aims at facilitating the integration of full-body control with virtual reality appli-

cations and video games when using OpenNI-compliant depth sensors and drivers. In June 2011, Microsoft released a non-commercial Kinect Software Development Kit (SDK) for Windows that includes Windows 7-compatible PC drivers for the Kinect device [33]. Microsoft's SDK allows developers to build Kinect enabled applications in Microsoft Visual Studio 2010 using C++, C# or Visual Basic. Microsoft is planning to release a commercial version of the Kinect for Windows SDK with support for more advanced device functionalities. There is also a third set of Kinect drivers for Windows, Mac and Linux PCs by the OpenKinect (libFree- Nect) open source project [34]. Code Laboratories CL NUI Platform offers a signed driver and SDK for multiple Kinect devices on Windows XP, Vista and 7 [35].

In this paper, we propose an automatic surveillance system for user identification and object recognition based on multi-modal RGB-Depth data analysis. We model a RGBD environment learning a pixel based background Gaussian distribution. Then, user and object candidate regions are detected and recognized using robust statistical approaches. On one hand, for the case of user identification, Random Forest using depth features are used to detect users, and RGB data is then used to identify the user combining body color modeling and face recognition. Face recognition in robustly performed based on Viola & Jones face detection, Active Shape Model alignment, face background substraction, SURF description, RANSAC outlier detection, and statistical learning of visual features. The system robustly recognize users and update the system in an online way, identifying and detecting new actors in the scene. On the other hand, segmented regions of candidate objects are described, matched, and recognized using view-point 3D descriptions of normal vectors using spatial and depth information, being robust to partial occlusions and local 3D viewpoint rotations. Moreover, 3D object information is online updated as well as new views of the object are detected. Finally, the system saves the historic of user-object pick ups assignments, being specially useful for surveillance scenarios. The system has been evaluated on a novel data set containing different scenarios, objects, and users, showing accurate recognition results.

The rest of the paper is organized as follows: Section 2 presents the novel surveillance system for user identification and object recognition based on statistical multi-modal data analysis. Section 3 presents the evaluation of the system on different scenarios. Finally, Section 4 concludes the paper.

2. Muti-modal user identification and object recognition

In this section, we present our system for automatic user-object interaction analysis using multi-modal RGBD data. The system is able to identify the subjects that appear in the scene as well as to recognize new or removed objects in the environment, controlling the membership of user objects based on the historic analysis of user-object interactions. The system is composed by four main modules which are described next: environment modeling, user detection and identification, object recognition, and user-object interaction analysis. All modules are integrated and work together in a robust multi-modal system for intelligent surveillance monitoring. The control automata of the system that calls to the different module functionalities is summarized in Algorithm1. The scheme of the whole system is illustrated in Fig. 1.

Data: $F_{\{1,..,T\}}$
Environment modeling of $F_{\{1,..,T\}}$ using pixel adaptive learning (section 2.1)
while *true* **do**
    Acquire new frame $F_t = \{I_t, D_t\}$ composed by RGB image I and depth map D (section 2.1)
    Segment new regions of F_t based on environment modeling (section 2.1)
    Look for subject/s and identification/s in F_t (section 2.2)
    Look for new objects or object removals in F_t (section 2.3)
    Look for getting/leaving objects in scene (section 2.4)
    User-object association analysis
end

Algorithm 1: Control automata of the RGBD surveillance system.

2.1. Environment modeling

Given a fixed RGBD camera, a background substraction strategy is applied in order to learn and adaptive model of the background in order to look for the presence or removal of elements in the scene and their posterior analysis by the different modules of the surveillance system.

Given the frame set $F = \{I, D\}$ containing a RGB image $I \in [0,1]^{h \times w}$ and a depth map $D \in [0,\infty]^{h \times w}$ with the depth value of each pixel obtained by the Kinect infrared sensor, an adaptive model is learnt for each pixel. Supposing a RGBD Gaussian distribution for each pixel, the training procedure is performed as,

$$\mu_{\mathbf{X},t} = (1-\alpha)\mu_{\mathbf{X},t-1} + \alpha\left(\frac{D_{\mathbf{X},t}}{\max D_t} \cup I_{\mathbf{X},t}\right), \tag{1}$$

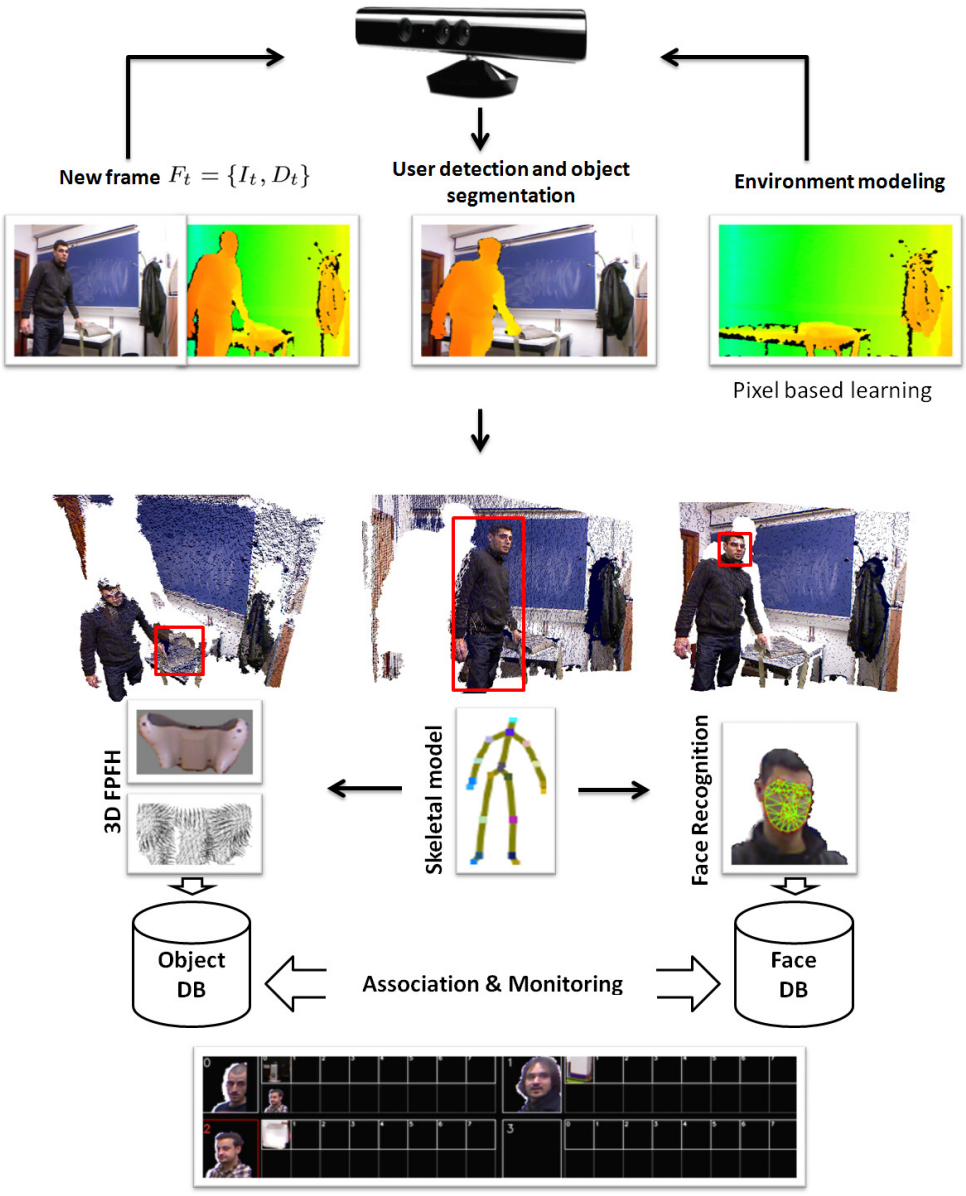

Figure 1: Muti-modal user identification and object recognition surveillance system.

$$\sigma^2_{\mathbf{X},t} = (1-\alpha)\sigma^2_{\mathbf{X},t-1} + \alpha \left(\frac{D_{\mathbf{X},t}}{\max D_t} \cup I_{\mathbf{X},t} - \mu_{\mathbf{X},t} \right)^T \left(\frac{D_{\mathbf{X},t}}{\max D_t} \cup I_{\mathbf{X},t} - \mu_{\mathbf{X},t} \right), \tag{2}$$

where $\mu_{\mathbf{X},t}$ is the mean depth learnt at pixel $\mathbf{x} = (i, j)$ at frame t, α is a training weight of the parameters during learning, $D_{\mathbf{X},t}$ is the depth at pixel $\mathbf{x}$ at frame t, $I_{\mathbf{X},t}$ is the RGB values at pixel $\mathbf{x}$ at frame t, and σ^2 is the covariance. The computation of μ and σ given a fixed α value is performed during a perfect stationary background composed of T frames, so that $t \in [1, .., T]$. Once the background has been modeled, a new change of a pixel in the scene produced by the appearance/disappearance of items is detected as follows,

$$\sigma_{\mathbf{X},T} - \left| \frac{D_{\mathbf{X},t}}{\max D_t} \cup I_{\mathbf{X},t} - \mu_{\mathbf{X},T} \right| > \theta_S, \tag{3}$$

where $|.|$ corresponds to the absolute value and θ_S is an experimentally set background segmentation hypothesis value. At the top of Fig. 1 one can see the background modeling procedure, a new frame F, and the detection of a new item corresponding to a user in the scene.

2.2. User detection and identification

Given the segmented image M that contains 1 at those positions satisfying Eq. 3 and 0 otherwise, the procedure for user detection and identification is only applied on the activated pixels of M. The algorithm for user detection and identification is summarized in Algorithm 2. The procedure is performed during n consecutive identifications to produce a final identification class or the detection of a new user in the environment. The value of n allows to prevent false isolate identification of users. Note that in our environment different users may appear in the scene at the same time, and thus, we track each particular user based on its distance to previous detections in time, as well as the counter for theT n identifications is treated for each user independently. Then, at each new frame, we perform user detection using the Random Forest approach with depth features of Shotton et. al [24] and compute the skeletal model. This process is performed computing random offsets of depth features as follows,

$$f_\theta(D, \mathbf{x}) = \mathbf{D}_{\left(\mathbf{x} + \frac{\mathbf{u}}{\mathbf{D_x}}\right)} - \mathbf{D}_{\left(\mathbf{x} + \frac{\mathbf{v}}{\mathbf{D_x}}\right)}, \tag{4}$$

where $\theta = (\mathbf{u}, \mathbf{v})$, and $\mathbf{u}, \mathbf{v} \in \mathbb{R}^{\mathbf{2}}$ is a pair of offsets, depth invariant. Thus, each θ determines two new pixels relative to $\mathbf{x}$, the depth difference of which accounts for the value of $f_\theta(D, \mathbf{x})$. Using this set of random depth features, Random Forest is trained for a set of trees, where each tree consists of split

and leaf nodes (the root is also a split node). Finally, we obtain a final pixel probability of body part membership l_i as follows,

$$P\left(l_i|D,\mathbf{x}\right)=\frac{1}{\mathcal{T}}\sum_{j=1}^{\mathcal{T}}P_j\left(l_i|D,\mathbf{x}\right), \tag{5}$$

where $P\left(l_i|D,\mathbf{x}\right)$ is the PDF stored at the leaf, reached by the pixel for classification $(D,\mathbf{x})$ and traced through the tree j, $j\in\mathcal{T}$.

Once the skeletal model is determined, we run the Viola & Jones face detector only in a reduced set of possible candidate regions around the head joint. Moreover, the detected face is aligned with the closed face of each possible user candidate in the data set using the mesh fitting error of an Active Shape Model procedure [36]. In case that no previous users exist, the data is saved as a new user with a new identifier. Then, a identification user procedure based on face description and body color modeling is applied to assign each of the n identification assignments, achieving the final identification by majority voting of the n intermediate results. Moreover, temporal coherence is taken into account by filtering the detections in time based on region density and 3D coordinates, discarding isolated detections and recovering miss-detections, resulting in a reduction of false detections and allowing a continuous detection of objects and users within the sequence. Next, we describe the procedure for user identification once a user has been detected and face is aligned with a particular class candidate.

2.2.1. User identification procedure

For the user identification module we propose to use the combination of body color model $\mathcal{C}$ with the face recognition probability $\mathcal{F}$ based on the matching of visual features, defining the following energy functional,

$$E(c_i,u)=\mathcal{C}(H_u,H_i)\cdot\beta+\mathcal{F}(f_u,f_i)\cdot(1-\beta) \tag{6}$$

where β is a trade-off energy parameter. Energy functional $E\in[0,1]$ is computed between a new test user $u=\{H_u,f_u\}$ and a candidate user class $c_i=\{H_i,f_i\}$, where H_i is the set of RGB color histograms for user i, and f_i is the set of face descriptions. Given a set of k possible users $C=\{c_1,..,c_k\}$ learnt online by the system, using the energy functional of Eq. 6, the new user candidate u is identified as follows,

$$\begin{array}{ll} i & \text{if} \quad E(c_i,u)>\theta_u, E(c_i,u)>E(c_j,u), \forall j\in[1,k], i\neq j \\ 0 & \text{otherwise} \end{array} \tag{7}$$

Data: $M_t, F_t, \mathsf{count}, n$
if *count* $< n$ **then**
  a) User detection [24] on D_t for the activated pixels in M
  if *Detected user* **then**
    b) Skeletal model description [24] on the pixels corresponding to the detected user
    c) Run Viola & Jones lateral and frontal face detectors on the surrounding areas to the detected head joint.
    if *Detected face* **then**
      d) Use Active Shape Model with a set of face landmark to align the detected face to the closest data set training sample for each subject based on the mesh fitting error
      e) Remove background from RGB aligned face for each possible user class
      f) Compute user body color histogram excluding face region (section 2.2.1)
      g) Perform user identification (section 2.2.1)
      h) Save the partial user identification ID_{count} to the class of the closest user probability, or 0 if none of the possible users achieve a probability threshold θ_u
      count++
    else
      count=0
    end
  else
    count=0
  end
else
  i) Assign class label to subject based on majority voting of ID or define new user if the majority vote is 0 count=0
end

Algorithm 2: User detection and identification algorithm.

In the case that the new user defines a new model (classification label 0), it is used to update the user model C with a new identifier $C = C \cup \{H_u, f_u\}$. In the case that the user has been identified as a previously learnt user, the user model can be updated if the energy E for the classified user is bellow a particular update threshold parameter, so that if $E(c_i, u) < \theta_u$ for the identified user i, then $c_i = \{H_i, f_i\} \cup \{H_u, f_u\}$, subtracting the oldest data to reduce an uncontrolled growing of model information. Next, we describe the computation of the color and face models.

Color model computation $\mathcal{C}$. Once a new user is identified in the environment, a predefined number of color histograms is defined, computed, and saved in the histogram set H_i for user i. Each histogram in this set is computed as a 62 bin normalized histogram (30-H and 32S) from HSV color representation (PDF of the HSV data for the subject) for each frame considered to model the user body color model, without considered the region of the subject detected as the face region. Once a new candidate user u is detected by the system, its color model histogram is computed and compared

with each learnt possible user i, defining the energy $\mathcal{C}(H_u, H_i)$ of Eq. 6. This energy is based on the Bhattacharyya similarity of two histogram distributions,

$$\mathcal{B}(h_u, h_i) = \sqrt{1 - \sum_j \frac{\sqrt{h_u^j \cdot h_i^j}}{\sqrt{\sum_j h_u^j \cdot \sum_j h_i^j}}} \tag{8}$$

where h_i^j is the j-th position of one of the histograms of the set H_i. Once this distance is computed among the candidate user u and each histogram in the training set, the m highest confidences of Eq.8 for each user class are selected to compute the mean confidence for that class. In this sense, the mean value reduces the bias that can be introduced by noise histograms for a particular subject in the data set. Thus, the final color energy term is defined as follows,

$$\mathcal{C}(H_u, H_i) = \frac{\sum_m \mathcal{B}(h_u, h_m)}{m} \tag{9}$$

for the m largest confidences for candidate user i.

Face model computation $\mathcal{F}$. Describing in more detail linez 7-10 of Algorithm 2, our steps for face model computation are,

• We perform face alignment using Active Shape Model by means of linear transformation of position, rotation and scale computed using the mesh fitting changes [36].

• We remove background from I on the region containing the face.

• We use fast SURF point detection and description on the RGB user face f_u and each candidate face f_i for user i [37].

• We match SURF features between f_u and f_i using nearest neighbor assignments. To increase robustness, matches are rejected for those keypoints for which the ratio of the nearest neighbor distance to the second nearest neighbor distance is greater than 0.69. It has been implemented using a k-d tree with Best-bin-first search [38].

• We use RANSAC to discard final outliers based on the difference of the pair of features assignment to the computed linear transformation. Inliers are selected based on linear least squares. If least than 8 correspondences are found the face is discarded.

• Using the initial set of v descriptions and the w final selected inliers, we compute a probabilistic membership of user model f_u to face model f_i for class i as follows [39]: Let $P(y|\neg f_i)$ be the probability that the matched features

y would arise by accident if the model f_i is not present. We assume the w feature matches arose from v possible features, each of which matches by accident with probability p. Therefore, we can use the cumulative binomial distribution for the probability of an event with probability p occurring at least w times out of v trials,

$$P(y|\neg f_i) = \sum_{j=w}^{v} \binom{v}{j} p^j (1-p)^{v-j} \tag{10}$$

To compute $P(f_i|y)$ we use Bayes' theorem,

$$P(f_i|y) = \frac{P(y|f_i) \cdot P(f_i)}{P(y|f_i) \cdot P(f_i) + P(y|\neg f_i) \cdot P(\neg f_i)} \tag{11}$$

We approximate $P(y|f_i)$ as 1 as we normally expect to see at least w features present when the model is present. We also approximate $P(\neg f_i)$ with the value 1 as there is a very low prior probability of a model appearing at a particular pose. Therefore, our face energy model $\mathcal{F}$ is computed as follows,

$$\mathcal{F}(f_u, f_i) = P(f_i|y) \approx \frac{P(f_i)}{P(f_i) + P(y|\neg f_i)} \tag{12}$$

As in the case of the color model $\mathcal{C}$, detected faces are used online to update the user model of faces either for the case of a new user or for the case of previously identified user. Figure 2 shows real application examples of the user identification approach based on the face energy $\mathcal{F}$.

2.3. Object recognition

Each segmented region (connected component) of M which has not been identified as a user is considered as a new object in case where the distance to the camera at those segmented pixels in D are reduced from the modeled background, or as the absence of an object if depth values increase.

The case where an object has been removed is straightforward to analyze since we saved the description of the object located at those positions from previous frame description. This means that if a user picks an object, we immediately know looking at the label of the object from the removed location which object it was.

In the case that a new object is located in a scene by a user, we take advantage of the 3D object information provided by the depth map D to

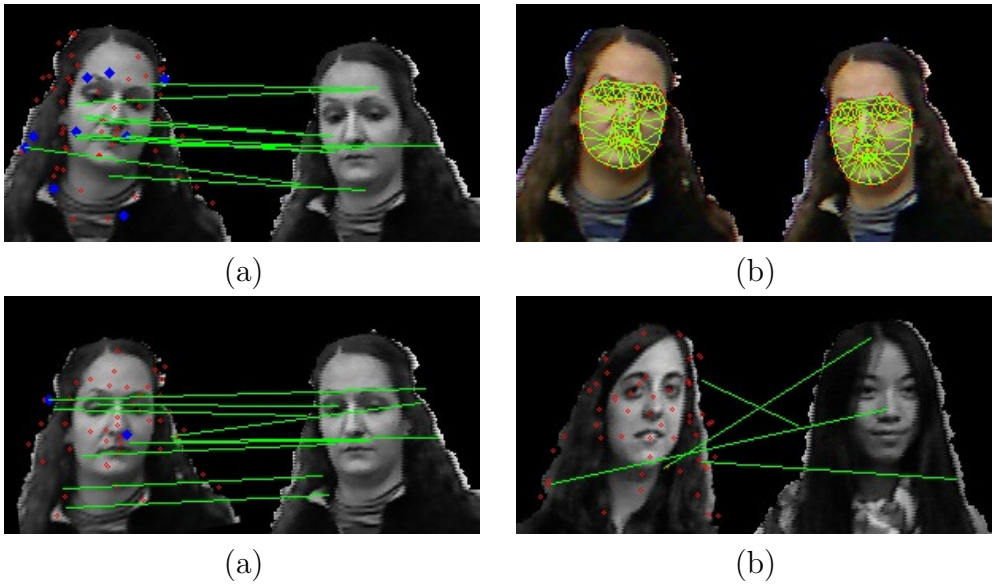

Figure 2: Face identification analysis (see better in color). Red dots: SURF candidate keypoints not matched based on descriptor distance. Blue dots: candidate keypoints discarted as outliers using RANSAC based on mesh transformation criteria. Green line: final matches considered for identification using Eq. 12. (a) Example of images not aligned after face detection and background removal. Several outliers are detected using RANSAC (blue dots), reducing final identification probability of being the same user category (71.4% of probability in this example). (b) Shows the intermediate results of applying ASM meshes to both faces before alignment. (c) Applying the whole proposed process. Now the probability of identification increases up to 98.4%. (d) An example of alignment and identification for two different categories, with a result of 32.3% of probability.

compute a normalized description of that particular 3D view [28]. For this task, we take use of the recently proposed Fast Point Feature Histogram (FPFH) of the Open Source Point Cloud Library (PCL) to compute a 3D rotation invariant object description for each particular point of view of an object $\mathcal{P}$ in the scene. A visualization of the descriptors for a set of objects is shown in Fig. 3. Given the depth information of an object for those pixels segmented in M, let define the object $\mathcal{P} = \{\mathbf{p}_1, .., \mathbf{p}_q\}$ as the set of 3D spatial coordinate point vectors $\mathbf{p}$. Then FPFH is basically computed as follows:

• In a first step, for each query point p_q a set of tuples ϕ, γ, δ encoding angular variations of the normal vectors among nearest points are computed, obtaining the Simplified Point Feature Histogram (SPFH).

• In a second step, for each point, its k neighbors are re-determined, and

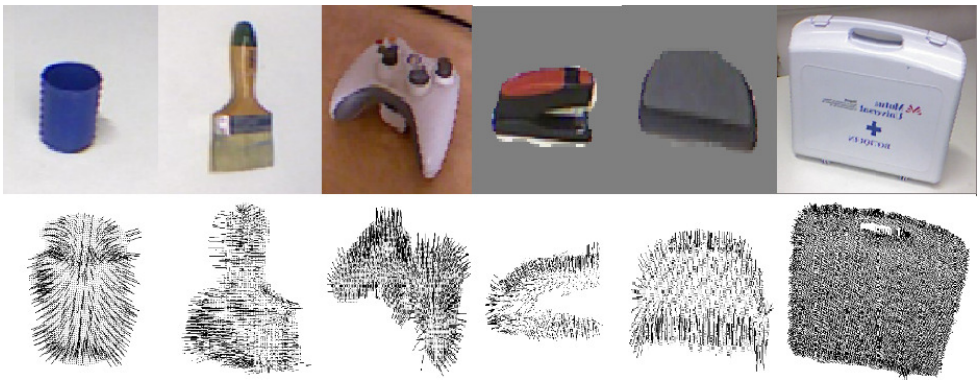

Figure 3: Views of different objects and descriptions based on the normal components.

the neighboring SPFH values are used to weight the final histogram of p_q (called FPFH) as follows:

$$FPFH(\mathbf{p}_q) = SPFH(\mathbf{p}_q) + \frac{1}{k}\sum_{i=1}^{k}\frac{1}{\omega_k}\cdot SPFH(\mathbf{p}_k) \tag{13}$$

where the weight ω_k represents a distance between the query point p_q and a neighbor point p_k, thus scoring the (p_q, p_k) pair [40].

The previous procedure is performed for each new object cluster in M, and the object description is compared to the data set of object descriptions saved in memory as in the case of the user color model $\mathcal{C}$. In this case, k-Nearest Neighbors are used to classify the new object view as a previous detected object if it achieves majority voting and a threshold value over object threshold θ_o, being also used to update online the data set of object descriptions. In cases where two objects achieve high similarity with the new sample, we update the model and fuse two previous object descriptions (i.e. object 1 and object 3 can be defined as different objects since their first appearance has been done by two completely different points of view. After that, the same object appears in the scene from an equidistance angle to object 1 and 3. As a consequence, we fuse objects 1 and 3 and update the model with the new point of view). An example of object segmentation and 3D visual description using FPFH is shown in the middle of Fig. 1 for a detected object in the scene.

2.4. User-object interaction

Given the user identification and object recognition proposed in previous section, the analysis of object-user interaction is straightforward. This step is based on the definition of pairs of values (user,object) for those new objects that appear in the scene or those users that pick up an object, looking for past memberships in order to activate the required surveillance alerts. Some interface examples are shown in Fig. 5 and 6 .

2.5. System discussion: computation complexity and accuracy

It is important to discuss that at each step of the surveillance system, the computational cost of each intermediate method has been analyzed and compared with different state-of-the-art approaches to define the most computationally feasible approach without a loss in generalization. Some particular details:

• Background modeling is learnt once, and then only one single operation (adaptive3) is performed per pixel at each frame in the segmentation step.

• User detection and skeletal model are only performed in the segmented frame regions, speeding up the procedure A_f/A_s times, where A_f is the area of the frame image and A_s is the area of the subject regions, and without any loss in accuracy.

• Temporal coherence is taken into account in the segmented regions among consecutive frames. If a region in consecutive frames do not achieve a 3D distance change of θ_T respect to an old segmented region, the analysis of the new segmented region is not performed since no significant differences in space are found, saving the computation of several analyses. In this case, we found that a good experimental set parameter θ_T achieves the same accuracy than analyzing all segmented regions in all frames and saves several object and user analysis per frame.

• In the case of user identification, using depth information to remove the background of the face region and compute posterior visual descriptors speeds up the procedure of face recognition up to 3 times. Moreover, we compared the same procedure with only RGBD data and obtained a significant loss in generalization.

Finally, though the system has not been parallelized, the iterative running of the approach in C++‘on a standard 2-CORE PC with 8GB of RAM computes 5FPS in mean. Since most of the modules are independent, parallelization can be easily applied, and other functionalities can be included, still being feasible for real-time surveillance purposes.

3. Results

In order to present the results of the proposed system, first, we discuss the data, methods and parameters, and evaluation measurements of the different experiments.

• **Data**. We defined a novel set of data recorded with the Kinect device[1]. The data set consists of 10 videos of one minute each one in indoor scenes and 5 videos of one minute each one in outdoor scenes. All videos are recorded at 24FPS, with a RGB resolution of 640 $\times$ 480 pixels, and a depth map of 320 $\times$ 280 pixels, calibrated and re-scaled to the RGB data size. The whole data set contains a total of 23600 semi-supervised labeled frames, containing a total of 8 different subjects and 11 different objects.

• **Methods and parameters**. We test the proposed system with the methods described in previous sections. The details of the parameter values of the approaches are: the depth threshold to limit the pixel analysis is fixed to 5 meters for accuracy purposes though the Kinect device can obtain high precision up to 10 meters. Trade off user identification threshold $\beta = 0.2$. Other face model parameters: mesh fitting error for convergence is 0.1, K-d tree for feature matching of 4 levels and assignments for values at least of 0.69, class probability for assign user is 97%, probability $p = 1/8$ prior for user identification in Eq. 10, and the number of consecutive user identifications for voting is 15. For object recognition, 5-NN is applied based on the FPFH values for matches of weight higher to 0.5. Each object or user candidate has to be maintained up to 5 frames to be analyzed and avoid false noisy detections. The historic of saved face descriptions, object descriptions, and user body color models has been fixed to a maximum of 40 instances, being removed the oldest ones when new data is updated. Minimum segmented area to analyze clusters is 200 pixels, and minimum real 3D distance considered for merging or considering different objects in the scene 6cm.

Moreover, we also compare the proposed system with state-of-the-art methods: SURF and Bag-of-visual-words (BOVW) description, and the effect of background substraction and face alignment for user identification. Finally we also compare with RGB SIFT description in the case of object classification.

• **Evaluation measurements**. In order to evaluate the proposed surveillance system, we used the novel labeled data set to compute the performance

[1]The data is public upon to request to the authors of the paper.

of the system in terms of user detection, user identification, object detection, object classification, user-object association, and theft. For each of these evaluations we measure the number of true positives, false positives, and false negatives. Moreover, we use statistical Friedman and Nemenyi test in order to look for statistical significance among the obtained performances [41].

3.1. Surveillance system evaluation

The mean global performance of the presented surveillance system is shown in Fig. 4. The Y-axis corresponds to the absolute value of true positives, false positives, and false negatives for each event category. One can see that we are able to correctly detect most of the events, corresponding to an accuracy upon 90%. Most true positives are detected. False positives are almost non existent except for the case of object detection, where small noisy regions of the image are sporadically detected as small objects. Only few false positives occur in the case of user identification and theft, where an error in the case of object or user detection/recognition immediately propagates an error in the final theft detection step.

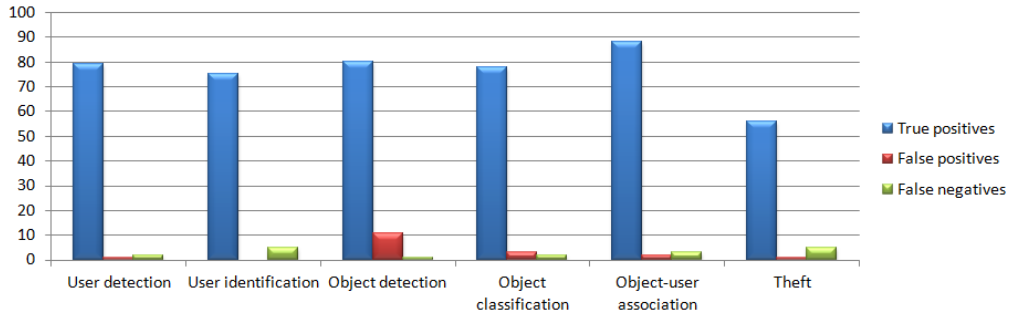

Figure 4: Mean surveillance system performance.

Some qualitative results of the execution of the surveillance system are shown in Fig. 5 and 6 .

3.2. User identification comparative

In order to compare the performance of the proposed system with standard approaches, in Table 1 we show the identification accuracy of our method (Statistical Surf) and the standard SURF description using Bag of Visual Words (SURF BOVW) [42] for the user identification module of our system. Moreover, for each of these two configurations, we test the effect

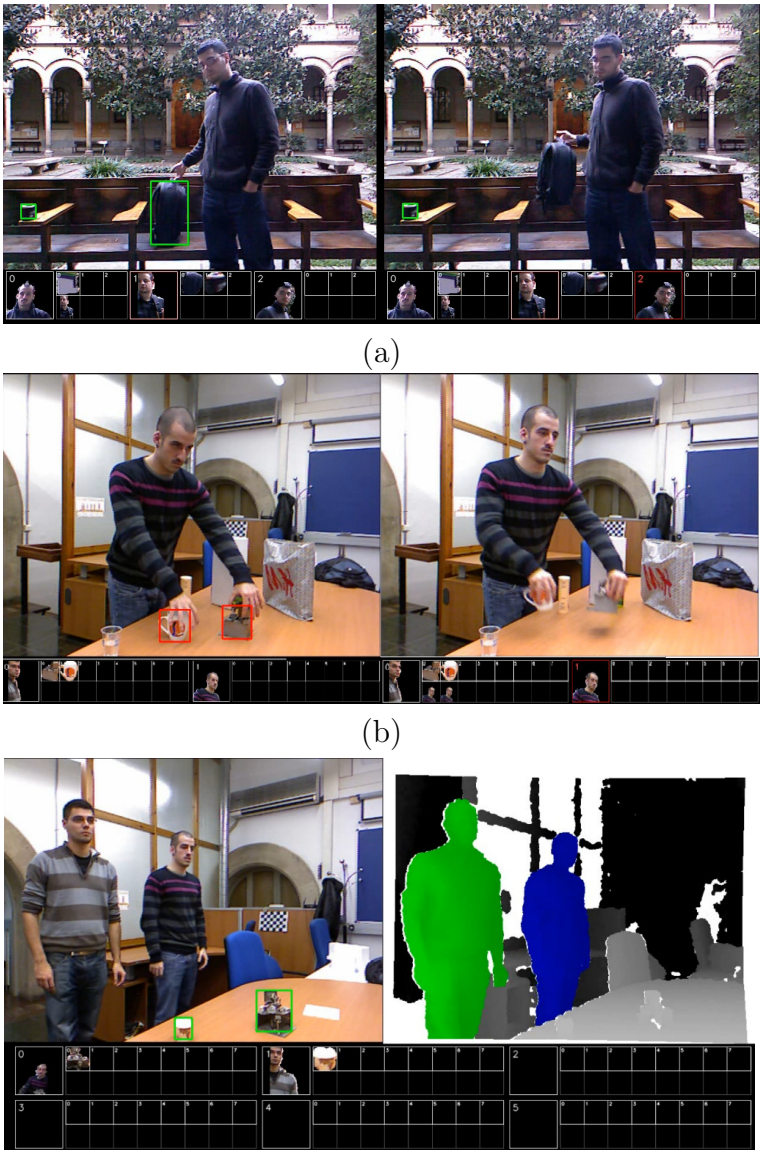

Figure 5: (a) Outdoor scenario: user is identified, theft is recognized, and different objects, included a small cup are detected. (b) Indoor scenario: simultaneous theft of two object by a user is correctly recognized. Several objects can be picked up or leaved simultaneously. (c) Users and object memberships are correctly identified and classified. Different users can be identified simultaneously by the system.

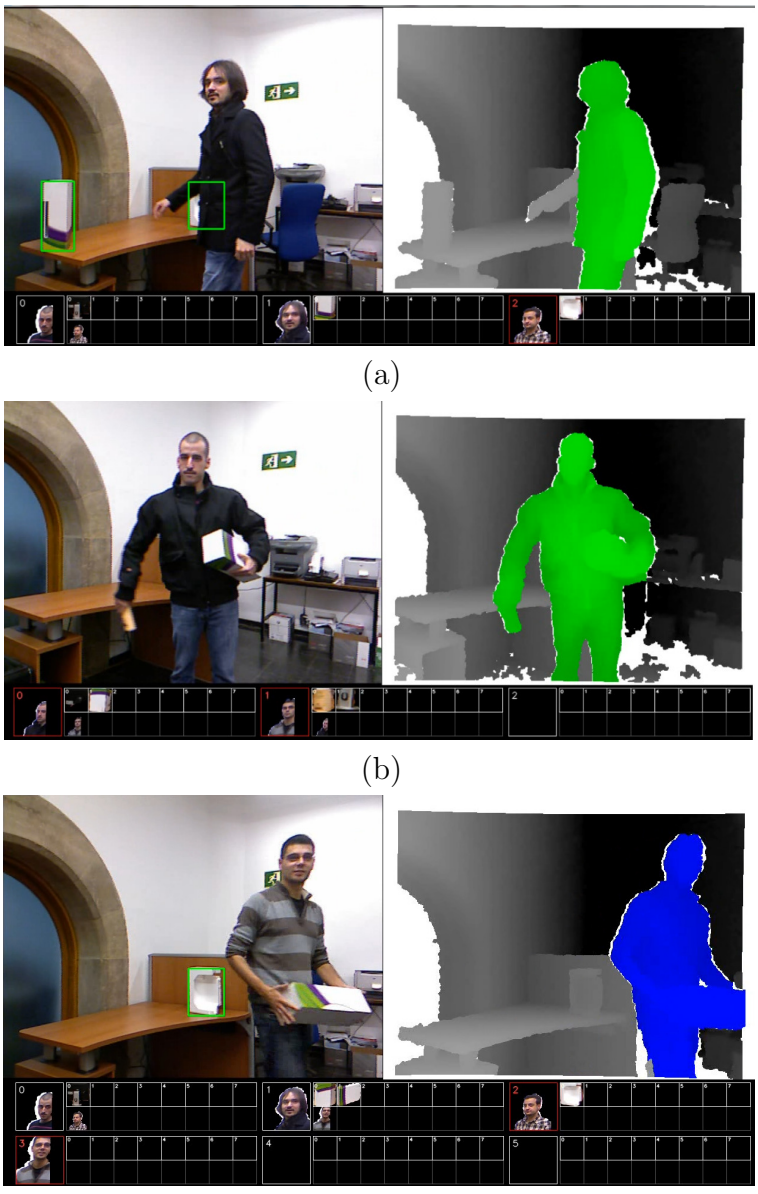

Figure 6: (a) Three identified users and two assigned objects. Object is identified even occluded by a user. (b) Thefts are detected. One user gets different objects, and different objects are associated to different persons. (c) Multiple users are identified by the system. A new user is identified and theft is correctly recognized.

SURF BOVW				STATISTICAL SURF			
$B + \overline{A}$	$B + A$	$\overline{B} + \overline{A}$	$\overline{B} + A$	$B + \overline{A}$	$B + A$	$\overline{B} + \overline{A}$	$\overline{B} + A$
33.3%	47.1%	52.8%	74.4%	52.9%	60.9%	76.3%	96.4%
7.2	6.2	5.3	2.6	5.2	4.4	2.4	1.1

Table 1: User identification performance results.

of removing background and aligning faces. In particular, A, $\overline{A}$, B, and $\overline{B}$ correspond to aligned, not aligned, with background, and background substraction, respectively. Comparing these approaches on the data set, one can see that removing background not only reduces the posterior complexity of the approach but also improves final identification performance. Aligning the face also increases the performance. Finally, one can see the robustness and better performance of our approach compared to the classical SURF BOVW technique, with a global mean improvement of 20% for the best configuration between both approaches on the presented data.

In order to compare the performances provided for each of theses strategies, the last row of Table 1 also shows the mean rank of each strategy considering the 15 different experiments. The rankings are obtained estimating each particular ranking r_i^j for each data sequence i and each system configuration j, and computing the mean ranking R for each configuration as $R_j = \frac{1}{N}\sum_i r_i^j$, where N is the total number of data sets.

In order to reject the null hypothesis that the measured ranks differ from the mean rank, and that the ranks are affected by randomness in the results, we use the Friedman test. The Friedman statistic value is computed as follows:

$$X_F^2 = \frac{12N}{k(k+1)}\left[\sum_j R_j^2 - \frac{k(k+1)^2}{4}\right] \tag{14}$$

In our case, with $k = 8$ system configurations to compare, $X_F^2 = 41.25$. Since this value is undesirable conservative, Iman and Davenport proposed a corrected statistic:

$$F_F = \frac{(N-1)X_F^2}{N(k-1) - X_F^2} \tag{15}$$

Applying this correction we obtain $F_F = 9.06$. With 8 methods and 15 experiments, F_F is distributed according to the F distribution with 7 and 98 degrees of freedom. The critical value of $F(7, 98)$ for 0.05 is 2.08. As the value of $F_F = 9.06$ is higher than 2.08 we can reject the null hypothesis.

RGB SIFT	DEPTH FPFH
86.2%	98.5%

Table 2: Object recognition performance results.

Once we have checked for the non-randomness of the results, we can perform an a post hoc test to check if one of the configurations can be statistically singled out. For this purpose we use the Nemenyi test. The Nemenyi statistic is obtained as follows:

$$CD = q_\alpha \sqrt{\frac{k(k+1)}{6N}} \tag{16}$$

In our case with $k = 8$ system configurations to compare and $N = 15$ experiments (data sets) the critical value for a 90% of confidence is $CD = 1.16$. As the ranking of the proposed probabilistic SURF approach with face background removal and aligned mesh does not intersect with any rank for that value of the CD, we can state that our proposal is statistically significant to the rest of system configurations in the presented experiments.

3.3. Object recognition comparative

In order to analyze the high discriminative power of the used FPFH descriptor encoding the normal vector distributions of a 3D object view, we compare the obtained recognition results with the standard object description using SIFT on the RGB segmented object region. The results are shown in Table 2. One can see that contrary to the state-of-the-art SIFT descriptor, the 3D-normal vector distributions improve classification results in 12% in the presented experiments.

4. Conclusion

We proposed an automatic surveillance system for user identification and object recognition based on multi-modal RGB-Depth data analysis. We modeled a RGBD environment learning a pixel based background Gaussian distribution. Then, user and object candidate regions were detected and recognized using robust statistical approaches. The system has been evaluated on a novel data set containing different indoor and outdoor scenarios, objects, and users, showing accurate recognition results. In particular, we showed that the proposed multi-modal methodology improves standard approaches

used in classical Computer Vision applications. As future work, we plan to analyze the reliability of the system to deal with smaller objects at different depth sizes, as well as to parallelize all modules of the system and apply it in real-time surveillance scenarios.

Acknowledgement

This work has been supported in part by the projects TIN2009-14404-C02, CONSOLIDER-INGENIO CSD 2007-00018, RecerCaixa 2011, and I+D+I IMSERSO 2011.

References

[1] N. H. W. S. A. J. Lipton, M. Allmen, Video segmentation using statistical pixel modeling.

[2] D. H. I.Haritaoglu, L. Davis, W4: A real-time system for detecting and tracking people in 2 d, internation Conference on Face and Gesture Recognition, Nara,Japan (1998).

[3] T. D. C. Wren, A. Azarbayejani, A. Pentland, Pfinder: Real-time tracking of the human body 19, iEEE Transactions on Pattern Analysis and Machine Intelligence.

[4] M. Isard, A. Blake, Icondensation: Unifying low-level and high-level tracking in stochastic framework. (1998) 893–9085th European Conference on Computer Vision, Freiburg, Germany: Springer Verlag.

[5] B. B. K. Toyama, J. Krumm, B. Meyers, Wallflower: Principles and practice of background maintenance (1999) 255–261International Conference on Computer Vision.

[6] H. F. A. Lipton, R. Patel, Moving target detection and classification from real-time videoIEEE Workshop on Applications of Computer Vision.

[7] D. H. A. Elgammal, L. Davis, Non-parametric model for background subtraction.6th European Conference on Computer Vision, Dublin, Ireland.

[8] D. H. T. Horprasert, L. Davis, A statistical approach for real-time robust background subtraction and shadow detection.Internation Conference of Computer Vision, Frame-rate Workshop.

[9] K. P. Karmann, A. Brandt, Moving object recognition using an adaptive background memory, in: Time-Varying Image Processing and Moving Object Recognition, V. Cappellini, ed. 2, Elsevier Science Publishers B.V., 1990, pp. 289–307.

[10] C. Stauffer, W. E. L. Grimson, Adaptive background mixture models for real-time tracking, Vol. 2, IEEE Computer Society, 1999, pp. 246–252.

[11] D. K. Howe, Active Surveillance Using Dynamic Background Subtraction, Tech. rep. (1996).

[12] W. Long, Y.-H. Yang, Stationary background generation: An alternative to the difference of two images, Pattern Recognition 23 (12) (1990) 1351 – 1359.

[13] A. Mittal, N. Paragios, Motion-based background subtraction using adaptive kernel density estimation, 2004, pp. 302–309.

[14] M. Cristani, M. Bicego, V. Murino, Integrated region- and pixel-based approach to background modelling, in: Proceedings of the Workshop on Motion and Video Computing, MOTION '02, IEEE Computer Society, Washington, DC, USA, 2002, pp. 3–.

[15] S. Huwer, H. Niemann, Adaptive change detection for real-time surveillance applications, in: Proceedings in Visual Surveillance, 2000, pp. 37–45.

[16] L. Li, W. Huang, I. Y.-H. Gu, Q. Tian, Statistical modeling of complex backgrounds for foreground object detection, IEEE Transactions on image processing 13 (11) (2004) 1459–1472.

[17] J. Connell, A. W. Senior, A. Hampapur, Y. l. Tian, L. Brown, S. Pankanti, Detection and tracking in the ibm peoplevision system (2005).

[18] A. Hampapur, L. Brown, J. Connell, A. Ekin, N. Haas, M. Lu, H. Merkl, S. Pankanti, Smart video surveillance: exploring the concept of multiscale spatiotemporal tracking, Signal Processing Magazine, IEEE 22 (2) (2005) 38–51.

[19] L. M. Brown, A. W. Senior, Y. li Tian, J. Connell, A. Hampapur, C. fe Shu, H. Merkl, M. Lu, Performance evaluation of surveillance systems under varying conditions, in: In: Proceedings of IEEE PETS Workshop, 2005, pp. 1–8.

[20] Y. Ivanov, A. Bobick, Y. A. Ivanov, A. F. Bobick, Recognition of multi-agent interaction in video surveillance, in: Internation Conference in Computer Vision, 1999, pp. 169–176.

[21] C. Stauffer, W. Eric, W. E. L. Grimson, Learning patterns of activity using real-time tracking, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (2000) 747–757.

[22] H. Z. Computer, H. Zhong, Detecting unusual activity in video (2004).

[23] R. Girshick, S. J., A. Criminisi, F. A., Efficient regression of general-activity human poses from depth images, IEEE International Conference of Computer Vision.

[24] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, A. Blake, Real-Time Human Pose Recognition in Parts from Single Depth Images, 2011.

[25] Y. Liu, C. Stoll, J. Gall, H. Seidel, C. Theobalt, Markerless motion capture of interacting characters using multi-view image segmentation, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011.

[26] S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. Davison, A. Fitzgibbon, Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera, in: Proceedings of the 24th annual ACM symposium on User interface software and technology, UIST '11, ACM, New York, NY, USA, 2011, pp. 559–568.

[27] A. Aldoma, N. Blodow, D. Gossow, S. Gedikli, R. Rusu, M. Vincze, G. Bradski, Cad-model recognition and 6 dof pose, in: ICCV 2011, 3D Representation and Recognition (3dRR11), Barcelona, Spain, 2011.

[28] R. B. Rusu, Semantic 3D Object Maps for Everyday Manipulation in Human Living Environments, in: Articial Intelligence (KI - Kuenstliche Intelligenz), 2010.

[29] Openni, http://www.openni.org.
URL `http://www.openni.org`

[30] Primesensor^{TM}.
URL `http://www.primesense.com/?p=514`

[31] Nite middleware.
URL `http://www.primesense.com/?p=515`

[32] Flexible action and articulated skeleton toolkit (faast).
URL `http://projects.ict.usc.edu/mxr/faast/`

[33] Kinect for windows sdk from microsoft research.
URL `http://research.microsoft.com/en-us/um/redmond/projects/kinectsdk/`

[34] Openkinect (libfreenect).
URL `http://openkinect.org/`

[35] Code laboratories cl nui platform - kinect driver/sdk.
URL `http://codelaboratories.com/nui/`

[36] T. F. Cootes, G. Edwards, C. Taylor, Comparing active shape models with active appearance models, in: in Proc. British Machine Vision Conf, BMVA Press, 1999, pp. 173–182.

[37] D. Kim, R. Dahyot, Face components detection using surf descriptors and svms, 2008 International Machine Vision and Image Processing Conference (2008) 51–56.

[38] D. G. Lowe, Distinctive image features from scale-invariant keypoints, International Journal of Computer Vision 60 (2004) 91–110.

[39] D. G. Lowe, Local feature view clustering for 3d object recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, 2001, pp. 682–688.

[40] R. Bogdan, N. Blodow, M. Beetz, Fast point feature histograms (fpfh) for 3d registration, in: The IEEE International Conference on Robotics and Automation (ICRA), Kobe, Japan, 2009.

[41] J. Demsar, Statistical comparisons of classifiers over multiple data sets, JMLR 7 (2006) 1–30.

[42] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, C. Bray, Visual categorization with bags of keypoints, in: In Workshop on Statistical Learning in Computer Vision, ECCV, 2004, pp. 1–22.