Journal of Medical Systems manuscript No.
(will be inserted by the editor)

Monitoring of medication intake using a camera system

Guillaume-Alexandre Bilodeau · Soufiane Ammouri

Received: date / Accepted: date

Abstract This paper presents a computer vision system for monitoring medication intake in the context of home care services. We use a method based on color and shape to detect the body parts and the medication bottles. Color is used for skin detection, and the shape is used to distinguish the face from the hands and to differentiate bottles of medicine. To track these objects, we use a method based on color histograms, Hu moments, and edges. For the recognition of medication intake, we use a Petri network and event recognition. Our method has an accuracy of more than 75% and allows the detection of the medication intake in various scenarios where the user is cooperative.

Keywords Medication intake monitoring · Face and hands detection · face tracking · hand tracking · Petri network

1 Introduction

Detection and tracking of body parts allow recognizing human activities remotely. Knowing that technology can significantly improve the life quality of elders and persons having mental illness, we are interested in the problem of monitoring medication intake, which is an activity involving the detection and tracking of the face and hands. Remote monitoring of patients allows them to stay at home longer and offers them flexible and efficient medical monitoring. The proposed system assumes that the patient is cooperative (does not try voluntarily to trick the system). Here, the system is designed to monitor the actual act of taking medications for patients that may suffer from distraction, temporary mental invalidity, or memory loss. For example, the patient might take out pills but then involuntarily not swallowing them, but thrashing them instead. A computer vision system would allow verifying such an event. The system uses one camera looking at a patient sitting on a table.

The design of such a camera based system involves many challenges. First, interactions between hands and medication bottles need to be detected. Then, interactions between

Guillaume-Alexandre Bilodeau · Soufiane Ammouri
Ecole Polytechnique de Montreal, P.O. Box 6079, Station Centre-ville, Montreal (Quebec), Canada, H3C 3A7
Tel.: 1-514-340-4711x5064
E-mail: guillaume-alexandre.bilodeau@polymtl.ca, soufiane.ammouri@polymtl.ca

the hands and the patient face (or mouth) need to be monitored. Finally, the pills themselves should be tracked. In this paper, we do not consider the tracking of the pills. Instead, interactions occurring between hands, face, and medication bottles are detected based on proximity and occlusions, and are the basis of the detection of medication intake. We believe this is a reasonable scenario for cooperative patients suffering from mental illness.

The main contribution of this work is selecting and adapting computer vision methods for medication intake monitoring. Furthermore, our implementation allows identifying precisely the difficulties for monitoring this activity. Our proposed method is the following. We initially use color to detect skin regions. Thereafter, a shape descriptor is used to locate and track the face. Hand tracking is done using the edges and centroid of the regions. For modeling medication intake, we use a Petri network. The token transition from a place to another occurs only if an event goes on for more than a specific duration. The detection and tracking of medication bottles are done by combining the color histograms and a shape descriptor.

The remainder of this document is structured as follow. In section 2, we present related works. Section 3 presents the methodology. Section 3.1 discusses initialization, section 3.2 presents the method used to detect the skin regions, and section 3.3 describes the methods used for detection and tracking of the face and hands. Section 3.4 presents the technique adopted for the detection and the management of occlusions between the body parts. In section 3.5, we explain how the medication bottles are located and tracked. Section 3.6 explains the Petri network developed for the detection of the human activity. In section 4, we provide the results and analyze them, and finally section 5 concludes the paper and presents some ideas for future work.

2 Related works

The field of video surveillance attracted a lot of interest, but it was little directed toward controlling medication intake. Recently, some papers (Batz et al, 2005; Valin et al, 2006) addressed this topic. In the work of Batz et al (2005), skin is first detected in each frame of the video using the *YCbCr* color space. Then, the bounding boxes of the skin regions and their centroids (center of mass) are used to detect occlusions between hands and the face. An occlusion is considered an interaction between two body parts. The shape and size of the regions are used to distinguish face from hands. The mouth is located based on the color of the lips, which in certain cases are significantly different from the face. This is not always the case. Medication bottles are detected in the image by searching for rectangular regions. Medication intake is detected by an event sequence composed, in order, of “opening medication bottle”, “a hand on the mouth”, and “closing the bottle”. This work relies on the orientation of the fingers to detect opening and closing of bottles. Fingers are not always visible, and this requires a precise tracking of the hands. Furthermore, the medication intake model does not allow changes in the order of events. A “closing the bottle” event before a “a hand on the mouth” event causes a misdetection of medication intake. They reported a recognition rate of 75% for medication intake.

In Valin et al (2006), three types of mobile regions are detected and tracked. They are the hands, the head, and the medication bottles. The authors use the algorithm which is described in Birchfield (1998) to track the head. In each image, a local search determines the ellipse who matches the best the head. To do so, this algorithm uses the intensity of the gradient, the perimeter of the ellipse, and the likelihood of skin color inside the ellipse.

Localization and tracking of the hands is performed by determining the skin regions that correspond to hands, based on some hypotheses, like the patient is wearing a long-

sleeved shirt. These hypotheses can be restrictive, but it avoids identifying the hand in the arm region. Interactions between hands and face are detected by counting the number is skin regions and by their positions. The two regions corresponding to the two hands become one region when they are occluded. Medication bottles are distinguished based on color (Habili et al, 2001).

Medication intake is expressed as scenarios based on events as done in Hongeng et al (2004). Examples of simple events are “one hand on the bottle”, “two hands on the bottle” and “one hand is getting closer to the face”. A second level of events (complex events) regroups simple events. For example, opening a bottle of medication is a sequence of “two hands on the bottle”, “one hand on the bottle”, “two hands on the bottle”, and “one hand on the bottle”. Finally, a scenario regroups complex events. They reported a recognition rate of 90% for medication intake.

Several researches have been done on the subject of object tracking. One possible way to track the hands and the face is to use statistical methods that optimize the appearance likelihood between the model and the target in the image either by sampling (Isard and Blake, 1998), or by a gradient descent (Comaniciu et al, 2003). In Jacquot et al (2005), the authors proposed a robust appearance model which combines color distributions and particle filters (Isard and Blake, 1998) to track non-rigid objects like faces, cars and football players. Rui and Chen (2001) proposed to track the face contour based on a particle filter also. In Comaniciu et al (2003), they used a special mask with an isotropic kernel and employed the mean-shift procedure to perform the tracking. With this type of methods, both hands can be confused after occlusion, since they both have the same appearance. Another possibility is to use a method based on bounding box overlap between two consecutive frames (Fuentes and Velastin, 2006). In this case, specific occlusion handling is needed.

In our case, since we are interested in particular events, body part tracking by proximity in each frame and redetection after occlusion are interesting possibilities since occlusion can be solved based solely on hand and face models after skin detection. No matter which object we want to track in a video sequence, we need a model to describe it. This model may contain a priori information about the object as well as information extracted from previous frames. The model can include descriptors of color, of texture, of shape or of any other type of primitives. As part of our work, we combined these descriptors to be able to locate and track the face and hands.

In contrast to previous works, we address the challenge of locating the hands inside larger skin regions, and we propose a simple, flexible and efficient model for recognizing medication intake.

3 Methodology

To detect events related to medication intake, we process each frame in the following way (Ammouri and Bilodeau, 2008). After initialization (section 3.1), we detect skin color in the *HSV* color space and construct skin regions (section 3.2). Then, we track the non-occluded body parts (section 3.3). Next, we verify the occlusions between skin regions to determine interactions between body parts (section 3.4). We identify the medication bottles and their occlusions with the hands (section 3.5). Finally, events are detected based on occlusions, and our Petri net is updated (section 3.6).

We have made the following assumptions:

- The medicine are in bottles, the patient take them always at the same place (sitting at a table in front of the camera), and there is no skin color in the empty scene (scene without patient);
- Only one person is in front of the camera during medication intake;
- Only medication bottles are on the table;
- The hands and face are not occluded at the beginning of medication intake, and the image is large enough so that some fingers can be seen;
- The patient is wearing some kind of top (shirt, bathrobe, etc.) to allow localizing the face and hands more easily. The cloths can have any color or texture as long as their color is different from skin;
- The lighting conditions are similar at each medication intake. Shadows are light.

For the camera setup, we envisioned different positions. We found that it was better to place the camera directly in front of the patient slightly higher than his head. In this way, hands, face, and medication bottles are more visible. This is a setup similar to previous work (Batz et al, 2005; Valin et al, 2006).

3.1 Initialization

In this proposed method, the system is initialized when three skin regions are detected. Initialization consists in detecting and locating in a single frame the body parts (head, hands) and the medication bottles. Then, tracking is performed. The next sections detail skin region detection and tracking.

3.2 Detection of skin regions

To detect the face and hands and to determine occlusions, we detect skin regions. Our selected method is based on thresholds in the *HSV* color space. The human skin is often represented by a portion of a particular color space, and it is therefore possible to extract color pixels which are similar to those of the human skin. The original video sequence is coded in the *RGB* color space. We begin by converting each frame of our input sequence into a color space more adapted to skin detection. We selected the *HSV* color space (Shapiro et al, 2001). The advantage of this color space is that it separates chrominance (color, H and S) from intensity V. Thus, hue H and saturation S become more independent of luminosity. We get a better robustness to light changes and shadows.

Some authors have proposed appropriate thresholds for skin detection operation (Sobottka and Pitas, 1996). In our work, we used the thresholds shown in Table 1 for the detection of skin pixels. For a pixel to be labeled as skin, it must be located within all intervals of Table 1. For H and S, we used the thresholds proposed by Sobottka and Pitas (1996), but for the intensity V, we considered only pixels that are not overly bright (reflections) or dark (shadows), as H and S have no meaning in these cases.

Once the classification of skin pixels is made, connected regions are found and small regions are removed. Each blob (region) contains information related to a cluster of pixels, such as bounding box, pixels list and area. The area property is used to reject blobs which are too small.

Figure 1 shows skin detection results. We can notice false detections for hair (Figure 1J), as well as undetected regions (e.g.: small parts of the forehead (Figure 1G)). In Figure 1E, the hair and the skin have similar color and therefore they are classified as skin regions. In

Table 1 Thresholds used for skin detection in HSV color space

Channel	Inferior threshold	Superior threshold
Hue	0 (0^o)	0.1 (36^o)
Saturation	0.2	1
Value	0.2	0.8

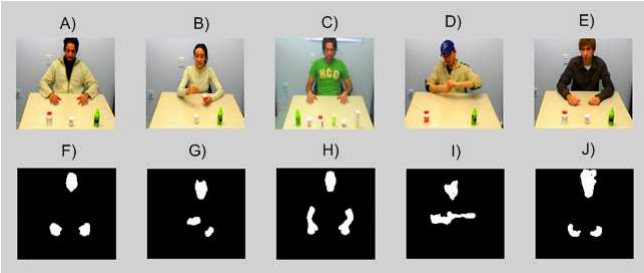


Fig. 1 Example of skin region detection. A), B), C), D), E) Frames of the original video sequences. F), G), H), I), J) Detected skin regions contained in each frame

the case of undetected regions, some regions reflect more light and thus appear to be more illuminated (overly bright regions). This change in the intensity of light modifies the color of the region in the *HSV* color space. The authors of Jacquot et al (2005) showed that when the saturation S is low and the value V is high or vice versa, the color tends toward white, thus escaping detection thresholds. To reduce the number of pixels that will not be detected due to reflections, we reduced the classification interval of V as shown in the table. In general, the thresholds chosen limit the conflict with other colors than those of the skin. The selected method can detect different types of skin with different conditions of illumination, and also can detect faces despite glasses or caps (Figure 1D). Furthermore, small amounts of additional or missing pixels do not change significantly the head shape. Thus, in the cases of figure 1, it has no impact as we do not need perfect body part segmentation, but their general shape.

The selected method has the advantages of speed and simplicity for skin region detection (Lemieux and Parizeau, 2002). Better algorithms exist (e.g. Baysian skin classifier (Kakumanu et al, 2007)), but the chosen method worked sufficiently well for our test videos. In any case, it could be replaced by any other skin detection methods if required. After thresholding and segmentation of skin regions, we can detect the face and hands in the source image. This step is important since a bad localization of the face and hands will decrease the possibility of recognizing human activity.

3.3 Body part tracking

3.3.1 Face detection and tracking

Once skin detection is done, we can isolate pixels that potentially belong to the same object (skin region). The purpose of this step is to detect and track the face separately from other body parts like the hands in this case. To do this, we suppose that a region R in our initial frame represents a face if (Batz et al, 2005):

1. The ratio between the width and the height of the region is smaller than T_{AL} ($T_{AL} = 2.25$ based on experiments). That is:

$$\frac{R_{MajAL}}{R_{MinAL}} < T_{AL}, \tag{1}$$

   where R_{MajAL} and R_{MinAL} are respectively the major axis length and minor axis length of R.
2. The ratio between the surface and the square of the perimeter of the region is larger than T_R ($T_R = 0.02$ based on experiments). That is:

$$\frac{R_{Area}}{(R_{Perimeter})^2} > T_R \tag{2}$$

   where R_{Area} and $R_{Perimeter}$ are respectively the area and perimeter of R.

The first test rejects regions which are too wide and narrow to be a face, such as a forearm. The second one is a measure of circularity, which identifies elliptical shape regions like the head. Note that the assumption of an elliptical shape for the face region is violated when there are occlusions, but as we will explain later, tracking is suspended in this case, so it is not problematic. This assumption can also be violated when the face changes its orientation. The authors of Batz et al (2005) assumed that face orientation should vary between 45° and 135°. This hypothesis is plausible if a head remains straight during some activity such as taking medication. We do not make this assumption because we want our algorithms for detection and tracking to be applicable to other human activities. Besides, this assumption is not needed for our method since the two tests above are used only in the initial frame (see section 3.1), not in tracking.

To track, we use another model for the face. Once the face is located in the initial frame of the sequence, we calculate the second order Hu moment for the corresponding blob (Choksuriwong et al, 2008). A Hu moment is an invariant shape descriptor that represents a sum on all pixels of the region weighted by polynomials related to pixel positions. Let us consider our image $I(i, j)$, where i and j are pixel positions after the extraction of skin regions. The moment of order $(p+q)$ for each blob is defined as

$$m_{pq} = \sum_i \sum_j i^p j^q I(i, j) \tag{3}$$

and central moment of order (p + q) as

$$\mu_{pq} = \sum_i \sum_j (i - \bar{i})^p (j - \bar{j})^q I(i, j) \tag{4}$$

with

$$\bar{i} = \frac{m_{10}}{m_{00}} \tag{5}$$

and

$$\bar{j} = \frac{m_{01}}{m_{00}}. \tag{6}$$

The normalized central moment of order (p + q) is defined as

$$\eta_{pq} = \frac{\mu_{pq}}{(\mu_{00})^\lambda} \tag{7}$$

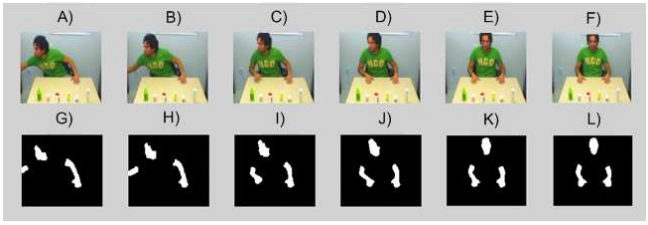


Fig. 2 Tracking of the skin regions. A), B), C), D) Frames 5, 8, 11, 14, 17 and 20 of the video sequence test. E), F), G), H) Detected skin regions contained in each frame

with

$$\lambda = \frac{p+q}{2} + 1. \tag{8}$$

These moments are invariant to translation and scaling. Hu moments are calculated from normalized moments and they are invariant to translation, rotation and scaling (Choksuriwong et al, 2008). We use the first and second order Hu moments:

$$\phi_1 = \eta_{20} + \eta_{02} \tag{9}$$

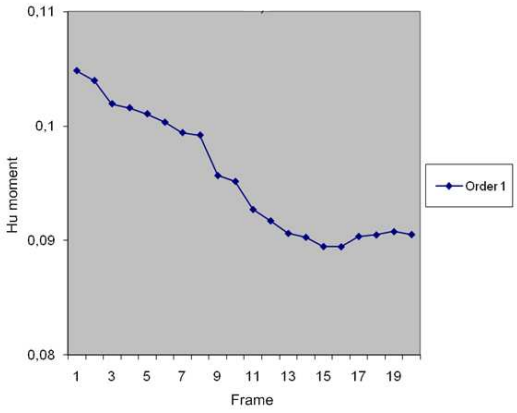
$$\phi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2. \tag{10}$$

After processing the initial frame, we calculate Hu moments of skin regions after the skin detection process. The idea of using a shape descriptor for tracking the face comes from the shape of the face which is different from hands and varies less in human activity. Face and hands have different Hu moments, and thus they can be distinguished. Tracking is performed by associating skin regions based on their Hu moments.

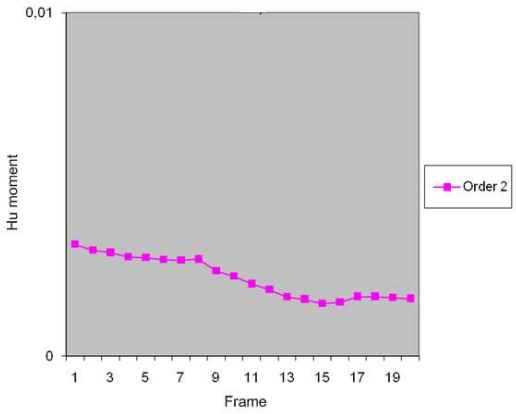
The use of Hu moments is justified by analyzing a sequence of frames in which the face changes orientation as shown in the figure 2. We examined the evolution of the Hu moments of order 1 (ϕ_1) and 2 (ϕ_2) for the blob corresponding to the face. Figure 3 shows that the Hu moment of order 1 is more sensitive to changes of face orientation compared to the order 2. Second order Hu moment allows also a good discrimination between face and hands as shown in figure 4. Consequently, it is the order 2 that is used for detection and tracking of the face. Tracking is done by comparing the second order Hu moment of each skin region with the identified face region in the previous frame. The region that is the most alike (smallest Euclidean distance between second order Hu moments) is considered the new face region.

3.3.2 Hands detection and tracking

After locating the blob which represents the face, the remaining skin regions are assumed to correspond to hands. To differentiate the two hands in each frame, we suppose the left hand is placed at the left of the right one, which is generally true. An analysis of blobs according to horizontal position allows us to differentiate between the two hands. Detecting and tracking hands become more difficult when the person wears a short-sleeved shirt. In this case, the hand must be located in the arm. The hand is located based on edge density. Because of the fingers, the edge density of the hand will be higher than for other part of the arm. This is why we require the camera to be close enough to make visible some boundaries between fingers.



(a) Order 1



(b) Order 2

Fig. 3 Comparison of the evolution of the Hu moments of order 1 and 2 of the face for the sequence of figure 2.

We developed an algorithm which is based on the contours to detect the hand. We use a Canny edge detector (Canny, 1986) followed by a dilatation to link edges and to obtain a better definition of the region contours. An edge detector detects and locates sudden changes of intensity in an image. Edge detection is applied on the binarized skin regions to obtain their contours. The result obtained with a contour detection using Canny method followed by dilation is presented in figure 5.

Once the skin region contours are found, we use a rectangular model for the hands. We use this model to obtain a bounding box that allows us to verify edge density. We do not need a precise model of the hand. The rectangular model for each frame of a video sequence is defined as

$$H = [B, L, C, F], \tag{11}$$

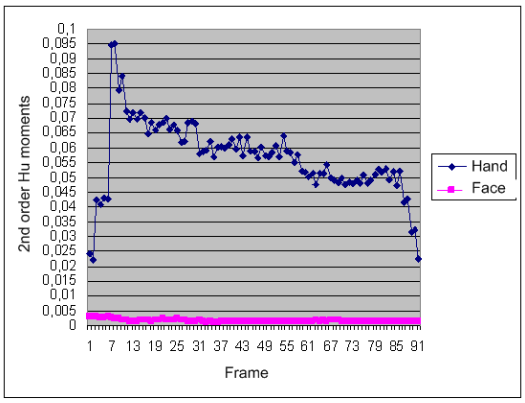


Fig. 4 Comparison of the evolution of the Hu moments of order 2 of the hand and the face for a sequence of frames

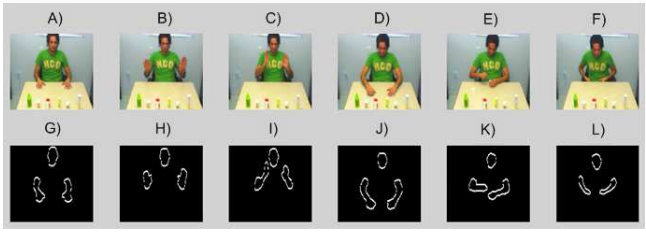


Fig. 5 Detection of contours of skin regions. A), B), C), D), E), and F) Frames 25, 125, 230, 325, 425 et 650 of the video sequence test, G), H), I), J), K), and L) Detection of contours with the Canny method followed by dilation

where B represents the smallest rectangle that includes the arm region (bounding box), C is the center of the rectangle representing the hand, L is the width of the hand and F is the density of hand edges. For each frame, we suppose that hand width is approximately 4/5 of the face width in the current frame. This has been determined experimentally. The hand width will be automatically adjusted if the user changes his position in relation to the camera because the face region size will also change.

Thus, for each skin region which does not represent the face, we start by locating both region extremities. This localization is performed by using the size of the bounding box and hand width, and the main orientation of the arm (horizontal and vertical). Figure 6 shows an example of candidate hand regions for the right arm.

The best candidate hand region is the one which contains more edges because of the fingers. For both rectangles, we calculate the edge density as defined in Shapiro et al (2001) with

$$F = \frac{\sum_{(i,k)\in R} I_c(i,j) | I_c(i,j) = 1}{N}, \tag{12}$$

where N is the number of pixels in each rectangle and I_c is the image containing the edges of the skin regions. Finally, we complete our hand model by calculating the center of the

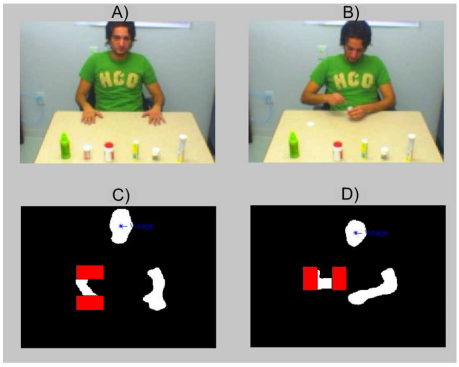


Fig. 6 Detection of the regions which can contain the hand in right arm A), B) Frames 25 and 425 of the video sequence test. C), D) Right hand candidates are the red rectangles

rectangle which covers the hand. From this model, tracking is performed in two ways: 1) By redetecting the hands at every N_f frames, and 2) by proximity with the previous hand positions. Tracking by proximity is done by comparing the centers of the two extracted rectangles which should include the hands of the current frame, with the previous hand positions. We supposed that for N_f frames, hands do not change enough shape which minimizes the tracking errors when comparing the model centers. In the event of errors, the following frame, in which we will calculate the edges density (redetection), will allow the system to catch up and relocate the correct hand position. Experimentally, N_f has been set to 6.

3.4 Occlusion between the skin regions

The detection of occlusions is used to detect events. For example, occluded hands are associated to the event “opening a bottle”. Thus, detecting occlusions is an important step in our system. Two approaches exist to handle occlusions in multiple object tracking: the merge-split approach and the straight-through approach (Gabriel et al, 2003). In the merge-split approach, occlusions between regions are detected, but individual region tracking is suspended. It means that we do not try to identify individual objects during occlusion. In other words, occluded objects are tracked as a group. In the case of the straight-through approach, objects are tracked individually during the occlusion. It requires some way to separate the occluded regions into individual regions corresponding to the tracked objects. In Lanz (2006), the author proposed to use color as a feature with a particle filter to be able to track several people. In our case, the tracked objects are parts of the body and do not have a discriminating color. In order to minimize false detections and errors in tracking of skin parts, the tracking of these regions is suspended for the duration of the occlusion. When objects are separated, tracking and monitoring is resumed.

Thus, we use the merge-split approach. We just need to know which two body parts are in occlusion to detect an event. Besides, because both occluded regions are of skin color, it is very difficult to separate them. Our strategy is to compute the number of skin regions detected to determine occlusion. Based on our assumptions, the maximum number of skin regions detected should be three. If it is less than three, there is an occlusion, and we just need to identify the involved body parts.

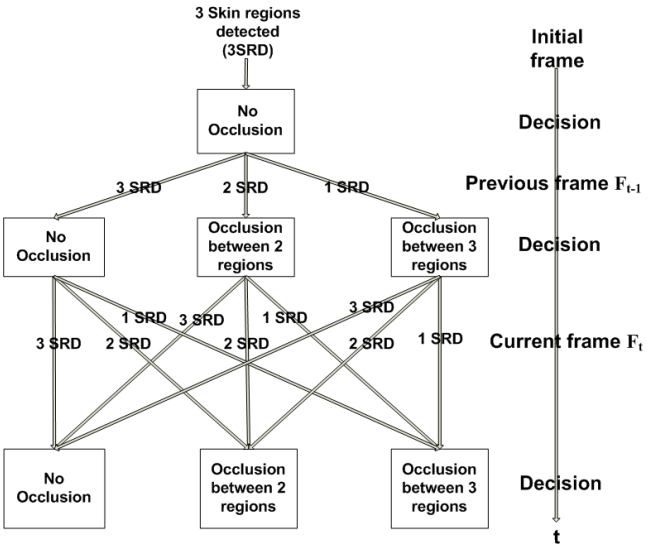


Fig. 7 Occlusion handling using the extracted skin regions. SRD: Skin regions detected

For handling occlusions, we assume that parts of the body (hands + face) are visible and they are not in occlusion in the initial frame. Thus, each region can be identified as hand or face. Thereafter, we examine the number of skin regions extracted in each frame to manage occlusions. In other words, we compare the number of skin regions detected in the current frame F_t with those in the previous frame F_{t-1}. This comparison allows us to identify the presence of occlusions and the number of occluded regions as shown in Figure 7.

It is important to mention that in this work, we do not use 3-D positions. When we mention occlusions, collisions are included. Without a three-dimensional positioning, we cannot distinguish between the two. Thus, once an occlusion is detected, we compare the distances between skin region centroids of previous frame F_{t-1} with the occluded region in F_t to determine which regions are now in occlusion. The detection of occlusions between skin regions allows us to detect the following events:

- Occlusion between left hand and right hand (Opening a bottle, if one hand is on the bottle);
- Occlusion between left hand and face (Swallowing pills with left hand);
- Occlusion between right hand and face (Swallowing pills with right hand).

3.5 Medication bottle detection and tracking

Our system is designed for medication bottles with different shapes and colors. To create the color and shape models of the medication bottles, a group of images which contains regions of each bottle is used for training. The color histograms and the second order Hu moments of these regions are calculated. We just need to locate the medication bottles in the image to detect them later on. Although a multiscale search everywhere in the image could be used, we decided instead to detect the table using its color. Thus, the color of the table on which the bottles will be put is also learned. This allow us to detect the table and to consider as candidate medication bottles, only regions included on the table area (see figure 8B)).

Fig. 8 Table detection. A) Original image B), Detected table

We assume that at the beginning of the video sequence, the bottles are not in occlusion. This allows us to know the initial number of bottles on the table. For all the objects on the table, we calculate their color histograms and their second order Hu moments. We thus detect the objects which have the same characteristics (color + shape) as the medication bottles learned. The color histograms are considered as vectors and the distance between them is calculated using the minimum cost to switch from one distribution to the other using the minimum distance of pair assignments (MDPA) distance (Cha and Srihari, 2002):

$$D(h(I),h(M)) = \sum_{j=0}^{K-1} \left| \sum_{k=0}^{j} (h(I)[k] - h(M)[k] \right|, \tag{13}$$

where $h(I)$ is an histogram from the current frame and $h(M)$ is a model histogram.

Figure 9 shows a medication bottles detection example. The bottles are well identified and they are not mixed up with the hand areas on the table. Tracking is done to detect the event “picking-up bottle”. This event is detected when a bottle is not on the table anymore. This condition is verified when the number of detected medication bottles is smaller than in the previous frame. Given that the bottles are rigid objects, we track them by comparing the centroids of the objects detected on the table for the current frame with the centroids of the medication bottles detected in the previous frame. For each frame, we calculate the number of bottles detected. If this number is lower than for the previous frame, we detect that a bottle is in occlusion or collision with a hand. Then, we calculate the distances between the centroids of the two hands and the centroid of the bottle taken to know which hand handles the bottle. In order to limit errors in detection and tracking of bottles, the tracking of the medication bottle is suspended for the duration of the occlusion. When the object is put back on the table, tracking is resumed.

This allow the detection of the following events:

- Occlusion between left hand and a medication bottle (Left hand picking-up bottle);
- Occlusion between right hand and a medication bottle (Right hand picking-up bottle);
- Laying down bottle (when it is put back on the table).

3.6 Human activity recognition

For medication intake modeling, we used a Petri network. A Petri net is an abstract model of the flow of information in a system (Peterson, 1977) and it is represented by a graph with two types of nodes (places and transitions) connected by arcs. A Petri net evolves when a transition is done: tokens are taken from places in the entry of this transition and sent in the exit places. A scenario is thus recognized when a token reaches the final place of the

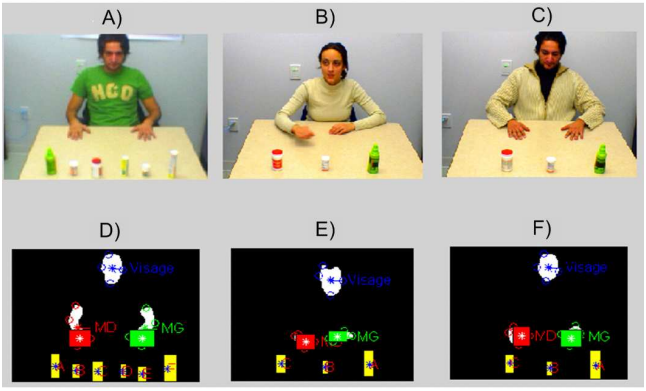


Fig. 9 Example of skin region and medication bottle detection. A), B), C) Frames of the original video sequences. D), E), F) Detected skin and bottle regions

Table 2 The minimum time duration of each event used in our work

Event	E1 E2	E3	E4 E5 E9 E10	E6 E7 E8
MTD (in seconds)	0.67	1.33	1.33	0.14

Petri net. The authors of Ghanem et al (2004) evoked the advantages of using Petri networks for the representation and the recognition of events. Among these advantages, we find the representation of the events in a sequential, simultaneous, and synchronized way. Our Petri net, designed for representing and detecting medication intake, has seven places (P1...P7) and ten transitions (E1...E10) as shown in figure 10. Transitions are triggered by the detected events of sections 3.4 and 3.5. The events that trigger transitions are:

- E1: Left hand picking-up bottle;
- E2: Right hand picking-up bottle;
- E3: Opening a bottle;
- E4 and E9: Swallowing pills with left hand;
- E5 and E10: Swallowing pills with right hand;
- E6, E7, and E8: Laying down bottle.

At first, a token is put in initial place P1. If events E1 or E2 occur, we move the token in the P2 place. We use the logical and temporal relations defined in Ghanem et al (2004) for the construction of our Petri network. Two arrows pointing to a place means a logical *OR*. The medication intake activity is recognized when the token reaches place P7. For the E3 event, we suppose that bottle opening occurs when the user is in possession of a bottle and when there is an occlusion or contact between the two hands. In Batz et al (2005), no constraint of duration is defined to carry out the transitions. In our work, to validate transitions and to avoid false detections of events, the transitions are validated only if their durations exceed a certain number of frames. The minimum time duration (MTD) of each event was determined experimentally and is presented in table 2.

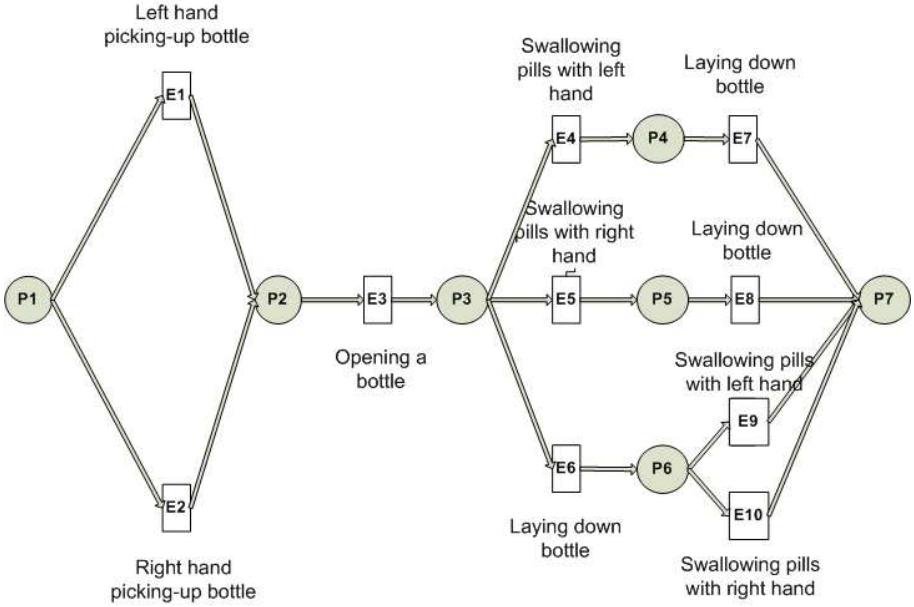


Fig. 10 Petri network used for the medication intake recognition

4 Experiments

In this section, we will present the results obtained on video sequences of medication intake scenarios. First, we present the experimental methodology.

4.1 Experimental methodology

No standard dataset is available. Thus, our method was tested with 320x240 video sequences acquired with a Sony DFW-SX910 camera. The subject is at about 2 meters from the camera (for fingers visibility, hands are about 35x35 pixels). Our method is implemented with Matlab. We validated three aspects of our method: 1) skin detection, 2) hands and face tracking, and 3) medication intake detection.

To test these aspects, we recorded twenty video sequences in which for twelve, an actor simulates medication intake, and in which for eight, an actor does another activity (e.g. working, eating, playing with medication bottles). For the twelve medication intake videos, the actors were asked to take medications as they would normally if they were sitting at a table. A ground-truth was created by labeling manually the hand and face regions in each frame for the twelve medication intake videos. Skin detection was validated by counting the number of frames in which there is false positive (FP_{sr}) and false negative skin regions (FN_{sr}). That is,

$$FP_{sr} = 100\% \times \frac{NF_{esr}}{NF_{sr}}, \tag{14}$$

and

$$FN_{sr} = 100\% \times \frac{NF_{msr}}{NF_{sr}}, \tag{15}$$

where NF_{sr} is the number of frames in the video with skin regions, and NF_{esr} and NF_{msr} are respectively the number of frames with extra skin regions and the number of frames with missing skin regions. Both values should be low.

The tracking performance was evaluated by counting the number of frames with true positive and false positive face and hands localization. That is,

$$TP_{face} = 100\% \times \frac{NF_{gfl}}{NF_f}, \tag{16}$$

$$TP_{hands} = 100\% \times \frac{NF_{ghl}}{NF_h}, \tag{17}$$

$$FP_{face} = 100\% \times \frac{NF_{ef}}{NF_{wf}}, \tag{18}$$

and

$$FP_{hands} = 100\% \times \frac{NF_{eh}}{NF_{wh}}, \tag{19}$$

where NF_f, NF_h, NF_{wf}, and NF_{wh} are respectively the number of frames with face, with hands, without face and without hands. NF_{gfl}, NF_{ghl}, NF_{wf}, and NF_{wh} are respectively the number of frames with good face localization, with good hands localization, with extra face, and with extra hands. The values for the first two metrics should be high, but should be low for the last two.

Finally, to evaluate medication intake detection, we used all twenty video sequences, and verified the Petri net transitions. Our system is successful if the pertinent transitions are activated based on the ground-truth events in the video. The efficiency E of our method was measured by

$$E = 100\% \times \frac{V_{gt}}{V_T}, \tag{20}$$

where V_{gt} is the number of video sequences with good transitions and V_T is the total number of video sequences.

4.2 Results and discussion

Table 3 shows the results we have obtained. For skin detection, we globally get low false positive and false negative values. Our algorithm had difficulties principally for the Guillaume and PierLuc video sequences. In the Guillaume sequence, reflections on the actor shirt created extra skin regions. For the PierLuc video, the actor's watch has split the arm region in half, and therefore created an extra skin region. False negatives were caused mostly by shadows.

For face and hands tracking, the results are again globally good. TP_{face} and TP_{hands} are high with values of 98% and 94% respectively. Note that when there are occlusions, if the hands and face were tracked correctly just before the occlusion, we considered that the tracking was good during the occlusion. In some cases, the hand may have a shape similar to

Table 3 Results of detection and tracking the body parts for 12 video sequences. *NF*: Number of frames in video

	FP_{sr} (%)	FN_{sr} (%)	TP_{face} (%)	FP_{face} (%)	TP_{hands} (%)	FP_{hands} (%)	Sleeve type	Glasses	*NF*
Sequence Francois	1	0	97	0	91	0	Short	Y	110
Sequence Soufiane1	0	3	97	0	98	0	Long	N	154
Sequence Soufiane2	0	1	99	0	97	100	Short	N	694
Sequence Atousa	0	0	98	0	99	0	Long	N	145
Sequence Soufiane3	0	0	96	0	93	0	Short	N	321
Sequence Karim1	0	0	97	0	92	0	Short	N	140
Sequence Karim2	0	0	100	0	100	0	Long	N	229
Sequence Ali1	0	0	100	0	95	0	Short	N	131
Sequence Ali2	0	0	100	0	100	0	Long	N	376
Sequence Younes1	0	0	97	0	92	0	Short	N	237
Sequence Guillaume	24	0	95	0	88	0	Long	N	141
Sequence PierLuc	25	0	100	0	73	0	Short	Y	208
Total	3	0.4	98	0	94	100			

the face. In this case, the second order Hu moments are almost similar and the system cannot classify the correct region of the face, and consequently, it commits an error in the detection of the hands. When the person is wearing a short-sleeves shirt, sometimes the hand may have an edge density lower than for the other extremity of the arm. This represents another source of errors for the system and its explains the lower accuracy of the system for hands compared to the face.. Furthermore, when a user wears a watch and has short sleeves, the arm and the hand will be split, and tracking will fail. In this case, the two parts should be merged to improve tracking, although this is not an easy task. In fact, Table 3 shows that our method performs better when the user wears long sleeves (99%, 99% versus 98%, 93% for face and hands, respectively). This is because the hand is easier to locate, and we do not need to search it in the arm.

We did not get any false positive for the face ($FP_{face} = 0\%$). However, for the Soufiane2 video sequence, we obtained a large false positive value for the hands ($FP_{hands} = 100\%$). This result is explained by the fact that we assumed that the face and hands are always in the image. In the case of the Soufiane2 video sequence, one hand gets out of the field of view of the camera for 15 frames. Thus, in these frames, the system detects an occlusion between the two hands because only two regions are visible (one hand + face).

As for glasses, they did not impact the performance of the system in the two videos featuring actors with glasses. As stated in section 3.2, we do not need the exact region corresponding to the face. We just need a good approximation of it (i.e. it must look like an ellipse more than the hands). Table 3 shows that bad skin detection has a significant impact on tracking (see PierLuc and Guillaume sequence results). Thus, although our skin detection method was adequate for most videos, a better method would improve tracking. Concerning the medication bottles, the system was able at any time to detect and track them correctly.

Since we have high TP_{face} and TP_{hands} values, we expect our method to be able to detect correctly medication intake. To further validate the performance of the detection and tracking part, we evaluated the performance of the medication intake detection. Figure 11 shows a graph (for the Atousa video sequence) which represents the periods when the principal events take place as well as the series of frames recorded by the system representing the duration of each event. Initially, the token is at P1. Event E2 occurs and it lasts long enough. The token is now at P2. Then, event E3 happens. The token is now at P3. Event E4 happens and the token moves to P4. Finally, event E7 occurs and the token is at P7. Medication intake is detected.

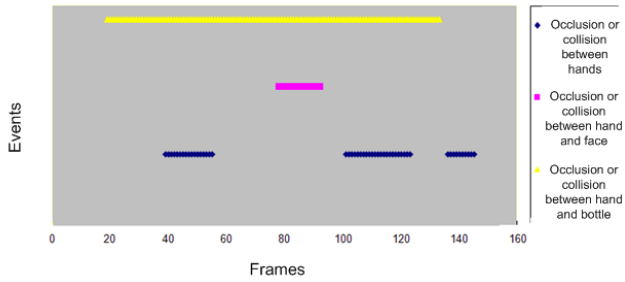


Fig. 11 Detected events in Sequence Atousa

Table 4 Results of detection and tracking the body parts for 10 video sequences

	V_{gt}	V_T	E (%)
Video sequences with medication intake	9	12	75
Video sequences without medication intake	2	8	25

Fig. 12 Low neck shirt enlarges the face area, and cause wrong event detection

Table 4 gives the performance for the twenty tested video sequences (12 with medication intake, and 8 with actors doing other activites). Our system has failed for three medication intake video sequences for the following reasons. In two cases, a hand is mistaken for a face. Thus, a two hands occlusion becomes a hand with face occlusion. If our method confuses body parts just before an occlusion, the wrong event will be detected. Another problem comes for the use of skin regions for tracking the body parts. As shown in figure 12, if the user wears a low necked shirt, this part of the body will be included with the face. Thus, occlusion between hands and face will be detected, even if they are not at head level. In this case, an elliptic model of the head could help to remove the neck area (Birchfield, 1998). For the videos without medication intake, there are two false negatives. In both case, after playing with medication bottles, the user puts a hand on his face. Since “Laying down bottle” is a valid event before swallowing the pills, this sequence of events corresponds to a medication intake activity. To remove such false negative, solutions would be to detect the pills, or to detect more precisely the positions of the fingers and the mouth. For activities like eating and working, the system did not detect medication intakes, because there were no interactions with the bottles of medicine.

Our method, being implemented in Matlab, is not real-time. It takes about 1.6 seconds to process a frame (see table 5). These times were obtained on an Intel Xeon processor running at 3.4*Ghz* and are an average for all tested video sequences. The most costly calculations are

Table 5 Typical processing time per frame

Processing step	time per frame (seconds)
Skin detection	0.1
Tracking face and hands and updating Petri net	0.6
Tracking medication bottles and updating Petri net	0.9
Total	1.6

the second order Hu moment and the MDPA distance. However, we did not do any particular efforts to improve processing time.

5 Conclusion

In this paper, we proposed a prototype system for detecting medication intake. Given some constraints on the medication intake environment, we proposed an algorithm that detects events based on occlusion between hands, face, and medication bottles. Body parts are tracked using skin detection and second order Hu moments. A Petri net was defined to model the medication intake activity based on detected events. It does not require any background extraction step and no specific training from the user. Our results show that we can detect medication intakes 9 times out of 12 and that we can track hands and face in more than 94% of the frames. Erroneous results are caused by skin detection which is not always as robust as required. A watch or a low necked shirt may cause body part detection errors, and thus tracking errors.

For future work, we will add additional features to detect the face and the hands. For example, we might consider an elliptic model of the head. This would improve body part detection and tracking. Also, localizing the mouth would improve medication intake detection accuracy. Furthermore, 3D positioning using stereoscopy for example, could reduce the number of erroneous contacts between objects thus allowing a more efficient management of collisions and occlusions. A localization algorithm for medication bottles that would allow an effective tracking, regardless of occlusions and contacts with the hand, would increase the confidence for the Petri net state transitions. Finally, an algorithm of face recognition could be added to our system to determine who is doing the activity.

References

Ammouri S, Bilodeau GA (2008) Face and hands detection and tracking applied to the monitoring of medication intake. In: Computer and Robot Vision, 2008. CRV '08. Canadian Conference on, pp 147–154

Batz D, Batz M, da Vitoria Lobo N, Shah M (2005) A computer vision system for monitoring medication intake. In: Computer and Robot Vision, 2005. Proceedings. The 2nd Canadian Conference on, pp 362–369

Birchfield S (1998) Elliptical head tracking using intensity gradients and color histograms. In: CVPR '98: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp 232 – 237

Canny J (1986) A computational approach to edge detection. IEEE Trans Pattern Anal Mach Intell 8(6):679–698

Cha SH, Srihari SN (2002) On measuring the distance between histograms. Pattern Recognition 35(6):1355 – 1370

Choksuriwong A, Emile B, Laurent H, Rosenberger C (2008) Comparative study of global invariant descriptors for object recognition. Journal of Electronic Imaging 17(2):1–35

Comaniciu D, Ramesh V, Meer P (2003) Kernel-based object tracking. Pattern Analysis and Machine Intelligence, IEEE Transactions on 25(5):564–577

Fuentes LM, Velastin SA (2006) People tracking in surveillance applications. Image and Vision Computing 24(11):1165–1171

Gabriel P, Verly J, Piater J, Genon A (2003) The state of the art in multiple object tracking under occlusion in video sequences. In: Proceedings of ACIVS, pp 377–380

Ghanem N, DeMenthon D, Doermann D, Davis L (2004) Representation and recognition of events in surveillance video using petri nets. In: Computer Vision and Pattern Recognition Workshop, vol 7, p 112

Habili N, Lim CC, Moini A (2001) Hand and face segmentation using motion and color cues in digital image sequences. In: Multimedia and Expo, IEEE International Conference on, pp 377–380

Hongeng S, Nevatia R, Bremond F (2004) Video-based event recognition: activity representation and probabilistic recognition methods. Computer Vision and Image Understanding 96(2):129 – 162, special Issue on Event Detection in Video

Isard M, Blake A (1998) Condensation – conditional density propagation for visual tracking. International Journal of Computer Vision 29(1):5–28

Jacquot A, Sturm P, Ruch O (2005) Adaptive tracking of non-rigid objects based on color histograms and automatic parameter selection. In: Motion and Video Computing, 2005. WACV/MOTIONS '05 Volume 2. IEEE Workshop on, vol 2, pp 103–109

Kakumanu P, Makrogiannis S, Bourbakis N (2007) A survey of skin-color modeling and detection methods. Pattern Recognition 40(3):1106–1122

Lanz O (2006) Approximate bayesian multibody tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence 28(9):1436–1449

Lemieux A, Parizeau M (2002) Experiments on eigenfaces robustness. In: Pattern Recognition, 2002. Proceedings. 16th International Conference on, vol 1, pp 421–424 vol.1

Peterson JL (1977) Petri nets. ACM Comput Surv 9(3):223–252

Rui Y, Chen Y (2001) Better proposal distributions: Object tracking using unscented particle filter. In: Computer Vision and Pattern Recognition, IEEE Computer Society Conference on, vol 2, pp 786–793

Shapiro LG, Stockman GC, Shapiro LG, Stockman G (2001) Computer Vision. Prentice Hall

Sobottka K, Pitas I (1996) Face localization and facial feature extraction based on shape and color information. In: Image Processing, 1996. Proceedings., International Conference on, vol 3, pp 483–486

Valin M, Meunier J, St-Arnaud A, Rousseau J (2006) Video surveillance of medication intake. In: Engineering in Medicine and Biology Society, 2006. EMBS '06. 28th Annual International Conference of the IEEE, pp 6396–6399