Machine Vision and Applications (2011) 22:219–234
DOI 10.1007/s00138-009-0234-7

ORIGINAL PAPER

A vision-based system for automatic hand washing quality assessment

David Fernández Llorca · Ignacio Parra ·
Miguel Ángel Sotelo · Gerard Lacey

Received: 28 August 2008 / Revised: 8 October 2009 / Accepted: 29 October 2009 / Published online: 18 December 2009

Abstract Hand washing is a critical activity in preventing the spread of infection in health-care environments and food preparation areas. Several guidelines recommended a hand washing protocol consisting of six steps that ensure that all areas of the hands are thoroughly cleaned. In this paper, we describe a novel approach that uses a computer vision system to measure the user's hands motions to ensure that the hand washing guidelines are followed. A hand washing quality assessment system needs to know if the hands are joined or separated and it has to be robust to different lighting conditions, occlusions, reflections and changes in the color of the sink surface. This work presents three main contributions: a description of a system which delivers robust hands segmentation using a combination of color and motion analysis, a single multi-modal particle filter (PF) in combination with a k-means-based clustering technique to track both hands/arms, and the implementation of a multi-class classification of hand gestures using a support vector machine ensemble. PF performance is discussed and compared with a standard Kalman filter estimator. Finally, the global performance of the system is analyzed and compared with human performance, showing an accuracy close to that of human experts.

Keywords Hand washing · Bi-manual gesture recognition · Kalman filter · Particle filter · Skin detection · Tracking · Multi-class SVM

1 Introduction

Effective hand washing is widely acknowledged to be the single most important activity for reducing the spread of infection in clinical environments and food preparation areas. It is extremely important that professionals use the correct technique to ensure that no areas of the hands are missed. According to several guidelines [20] the correct hand washing procedure should comprise six different poses as depicted in Fig. 1.

In the last few years, several authors have addressed the problem of single-hand gesture recognition [21], and recently gesture recognition has been applied to hand washing [9]. Among the wide range of vision-based hand gesture recognition techniques in the literature, we emphasize histograms of oriented gradients (HOG) which have been used as feature vectors in hand gesture classification [14]. Spatio-temporal hand gesture recognition using neural networks has been introduced in [15,27], and recognition of gestures using hidden Markov models (HMM) has been also widely studied [18,21,25]. In [19], a robust hand posture detection method based on the Viola and Jones detector [29] is proposed. In [8], the hands and the towel are modeled as a *flock* of features describing its approximate shape and three independent particle filters (PFs), one for each of the right and left hands, and one for the towel, are used.

Electronic supplementary material The online version of this article (doi:10.1007/s00138-009-0234-7) contains supplementary material, which is available to authorized users.

D. F. Llorca (✉) · I. Parra · M. A. Sotelo
Department of Electronics, University of Alcalá, Madrid, Spain
e-mail: llorca@depeca.uah.es

I. Parra
e-mail: parra@depeca.uah.es

M. A. Sotelo
e-mail: sotelo@depeca.uah.es

G. Lacey
Department of Computer Science, Trinity College Dublin,
Dublin, Republic of Ireland
e-mail: gerard.lacey@cs.tcd.ie

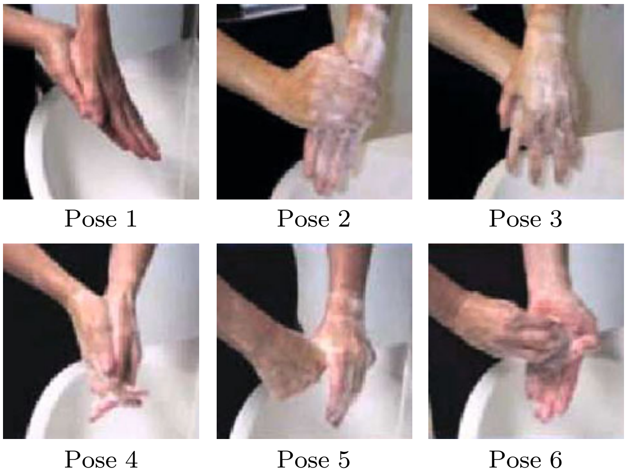

Fig. 1 *Pose 1* Rub palm to palm. *Pose 2* Rub palm over the back of the other one. *Pose 3* One hand over back of the other hand and rub fingers. *Pose 4* Rub palm to palm with fingers interlaced. *Pose 5* Wash thumbs of each hand separately using rotating movement. *Pose 6* Rub finger tips against the opposite palm using circular motion

In this work, a novel system for hand washing quality assessment using computer vision is proposed in the context of the Kinometrics project [13] whose main goal is to develop low cost wireless sensors to measure human movement. Hands/arms segmentation is achieved by combining skin and motion features to ensure a robust region of interest (ROI) selection process which is independent of lighting conditions or reflectivity of the sink (ceramic, stainless steel, etc.). Hands/arms are modeled by using an area-based ellipse fitting method. A single multi-modal distribution is then used in order to measure the position and orientation of both hands/arms [1]. A bi-manual gesture recognition scheme that combines appearance descriptors extracted from a single frame of a video sequence with motion descriptors extracted from optical flow is proposed. Specifically, HOG [6] and the so-called histograms of oriented optical flow (HOF) are used to create the feature vectors which will be dispatched to the classifier. In order to recognize the six different poses of the correct hand washing technique, two independent support vector machine (SVM) [28] classifiers with a one-against-one multi-class approach are used and combined in a second stage. Finally, a high-level analysis is carried out to measure the time spent in each pose by the user. Thus, it is possible to measure the quality of the hand washing. A general view of the proposed system is depicted in Fig. 2.

The remainder of the paper is organized as follows: Section 2 provides a detailed description of the hands/arms segmentation process along with the proposed tracking model. PF is widely described in Sect. 3. The feature extraction method, the description of the classifier used to recognize the different poses and the high level analysis used for the measurement of hand washing quality are described in Sect. 4. The results achieved up to date and their comparison with human performance are presented in Sect. 5. Finally, conclusions and a description of the future lines of research are presented in Sect. 6.

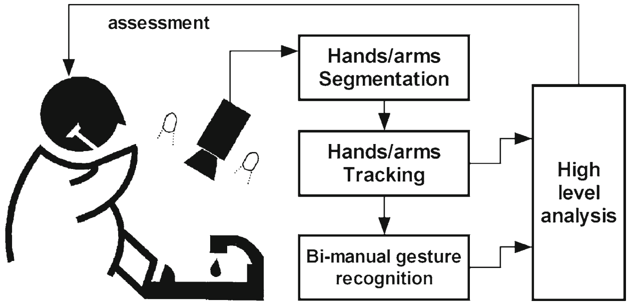

Fig. 2 General outline of the proposed hand washing quality assessment system

2 Hands/arms motion model and measure equation

2.1 Adapting vision measurements to the proposed model

2.1.1 Skin color detection

An area-based ellipse fitting over a skin probability map is used to measure the hands/arms position and orientation. A skin color segmentation method is used to generate the skin probability map. The aim is to have a probability map with high intensity values in the skin pixels and low intensity values in the non-skin pixels. Skin detection plays an important role in various applications such as face detection, searching and filtering image content on the web, video segmentation, face/head tracking, etc. Among the different color spaces we use the normalized RGB color space, which is easily obtained from the RGB values by a simple normalization procedure:

$$r = \frac{R}{R+G+B}, \quad g = \frac{G}{R+G+B}, \quad b = \frac{B}{R+G+B} \tag{1}$$

The normalization removes intensity information, so that rgb values are pure colors. Because $r + g + b = 1$ no information is lost if only two elements (r, g) are considered. In that case, the color space is usually named as *rg-chromaticity*. In the work described in [26] the illumination influence over the skin-tone color for several nationalities is studied using four fluorescent lamps with different CCTs. Thus trapezoidal areas in the rg-chromaticity plane group the different skin color values of the different subjects nationalities according to the illumination conditions [26]. In our approach skin segmentation is applied using rectangular areas in the rg-chromaticity plane instead of trapezoidal ones, since it facilitates the probability computation of the probability without loss of

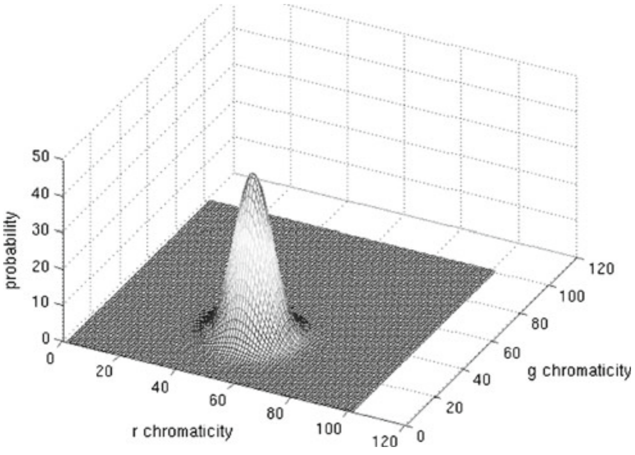

Fig. 3 Gaussian distribution in the rg-chromaticity plane

performance. Four boundaries are defined ($r_{\min}, r_{\max}, g_{\min}$ and $g_{\max}$) and the skin probability is then modeled by a Gaussian function:

$$f(r, g) = \frac{1}{2\pi\sigma_r\sigma_g} \exp(-\frac{(r - r_{\text{mean}})^2}{2\sigma_r^2} - \frac{(g - g_{\text{mean}})^2}{2\sigma_g^2}) \quad (2)$$

where $r_{\text{mean}}, g_{\text{mean}}, \sigma_r$ and σ_g are the mean and the variance values for each chromaticity channel, respectively. After some experimental work, the variances are fixed to a value of 0.6. r_{mean} and g_{mean} are computed according to the rectangular boundaries:

$$r_{\text{mean}} = \frac{(r_{\max} - r_{\min})}{2}, \quad g_{\text{mean}} = \frac{(g_{\max} - g_{\min})}{2} \quad (3)$$

If the chromaticity of a pixel falls in the modeled area (see Fig. 3), then its probability is computed using Eq. (2) and normalized between 0 and 255. Thus the intensity of the result images represents the probability of a pixel of being skin.

Although the described scheme can successfully deal with lighting changes, the varying lighting conditions of the wash hand basin requires a lightning compensation step to achieve more robust measurement of skin tone. We chose the grey-world approach implemented in [4]. This method is based on the assumption that the spatial average of surface reflectance in a scene is achromatic. Since the light reflected from an achromatic surface is changed equally at all wavelengths, it follows that the spatial average of the light leaving the scene will be the color of the incident illumination. In addition, even though lighting compensation reduces the variability in skin tones detected, there remains a significant challenge to robust segmentation in real world situations, i.e., reflections due to stainless steel wash hand basins. This is a factor of both the skin tones of the user and the lighting conditions. An example of this can be observed in Fig. 4. In Fig. 4a, reliable segmentation is achieved, while in Fig. 4b additional light from a window produces reflections of the hands on the stainless steel. Consequently, in situations where stainless steel wash hand basins are used, additional features are required to achieve robust segmentation. In our approach, we propose the use of features based on motion analysis.

2.1.2 Skin and motion segmentation

Our proposal for robust hand segmentation consists in computing an overall image that integrates both skin and motion features. The first frame used in the initialization step

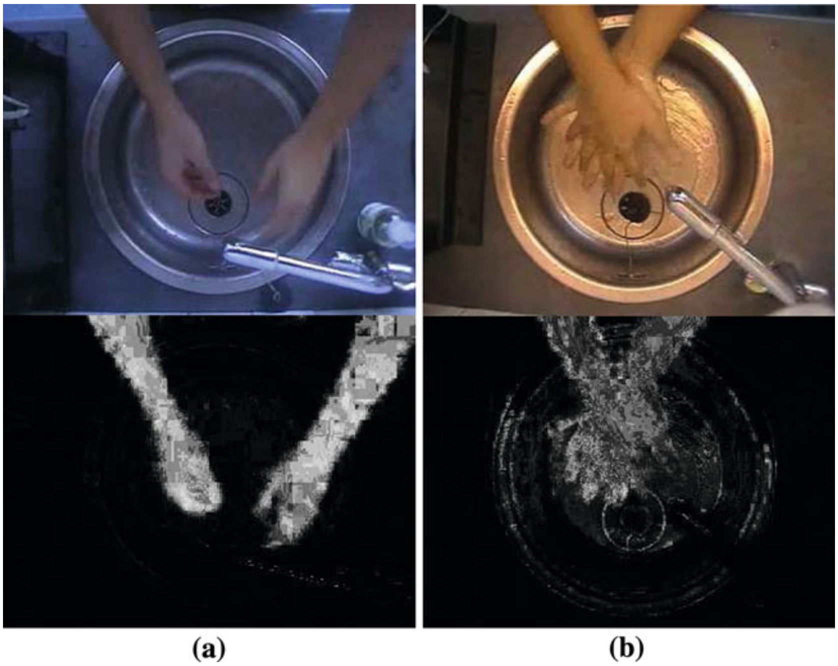

Fig. 4 *Upper row* Original images. *Lower row* Skin probability maps. **a** With favorable, and **b** unfavorable lighting conditions

is directly given by the skin probability map after skin color segmentation. Then, for each skin pixel, which could be a false positive, we calculate the motion vector obtained by means of optical flow analysis. Then, an average filter over the motion magnitude image is computed. If the averaged motion magnitude is greater than a fixed threshold T_{ID} the probability of a pixel to be considered as hands/arms region is increased. If it is less than T_{ID}, then the probability is decreased. Let $M_{i,j}$ represents the averaged motion magnitude of pixel (i, j) and $P_k(i, j)$ be the probability of pixel (i, j) to be considered as hands/arms region at frame k. Then

$$\begin{cases} \text{if} & M_{i,j} \geq T_{ID} & \text{then } P_k(i,j) = f_I(P_{k-1}(i,j), I) \\ \text{else if} & M_{i,j} < T_{ID} & \text{then } P_k(i,j) = f_D(P_{k-1}(i,j), D) \end{cases} \tag{4}$$

where f_I and f_D functions are given by

$$f_I(P_{k-1}(i,j), I) = \begin{cases} 1.0 & \text{if } P_{k-1}(i,j) + I > 1.0 \\ P_{k-1}(i,j) + I & \text{otherwise} \end{cases} \tag{5}$$

$$f_D(P_{k-1}(i,j), D) = \begin{cases} 0.0 & \text{if } P_{k-1}(i,j) - D < 0.0 \\ P_{k-1}(i,j) - D & \text{otherwise} \end{cases} \tag{6}$$

We implement this increase/decrease effect by means of the coefficients I and D, defined according to the following equations:

$$I = I_F\left(1 - \frac{1}{\exp\left(\frac{M_{i,j} - T_{ID}}{0.5T_{ID}}\right)}\right); \quad D = D_F \frac{1}{\exp\left(\frac{M_{i,j}}{0.5T_{ID}}\right)} \tag{7}$$

where I_F and D_F are the increase and decrease factors, $M_{i,j}$ is the motion magnitude of pixel (i, j), and T_{ID} is the threshold value which decides if there is going to be an increase or a decrease in the output image. T_{ID} is experimentally obtained taking into account possibly camera movements.

The increase function should reach the maximum value quickly, whereas the decrease function must be less steep. This is because we want to quickly highlight motion as soon as it is detected and penalize its absence more slowly. Thus skin pixels due to metal sinks and specular reflections are slowly removed while the hands/arms are correctly segmented only if they are moving. This architecture does not resolve the issue of hand detection while the hands are not moving. However, in this application, the hands rarely stop moving and one can argue that if they do stop no effective hand washing is being performed. Figure 5 shows a summary

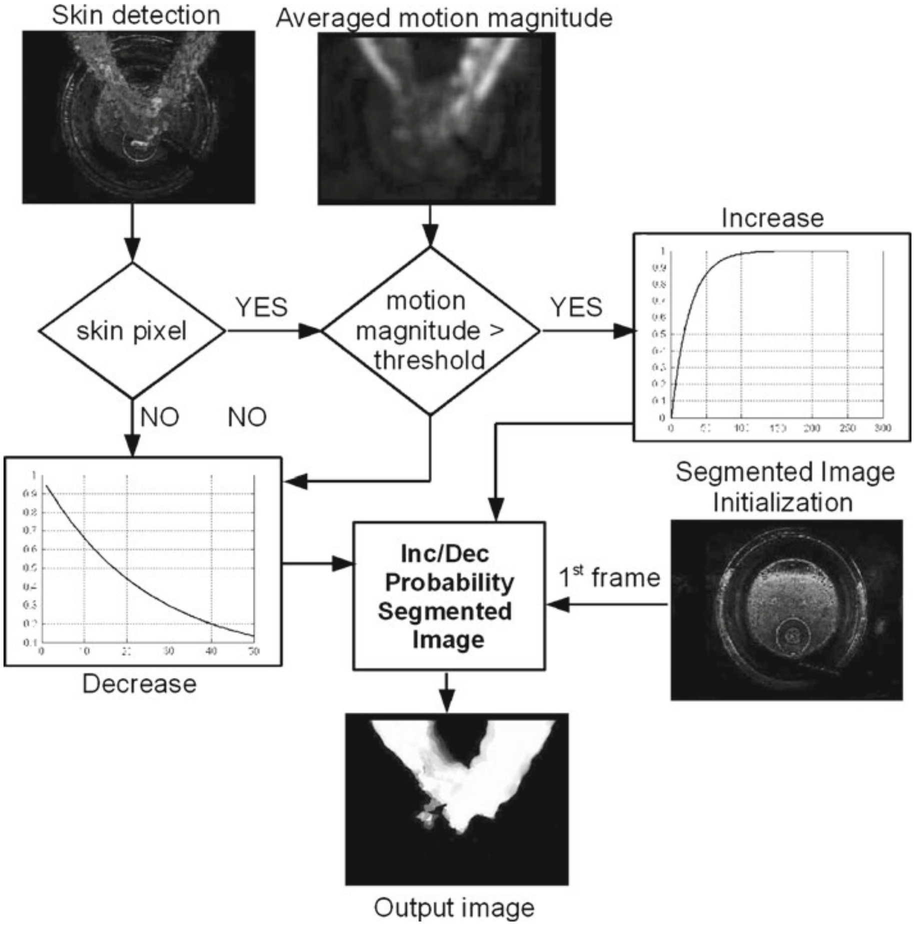

Fig. 5 Skin and motion detector for hands/arms segmentation scheme

of the global architecture described so far for the hands detection based on both color and motion information.

2.2 The model equations

The objective of the proposed tracking method is to model the hands/arms movements of a person washing their hands. In order to achieve this goal each one of the hands/arms has been characterized as an ellipse (Fig. 6) represented by the following state vector:

$$\mathbf{x}_k = \{x_k, y_k, \theta_k, \dot{x}_k, \dot{y}_k\} \tag{8}$$

where $x_k, y_k, \dot{x}_k, \dot{y}_k$, are the ellipse position and velocity and θ_k is its orientation. The ellipse axes were not included in the state vector because they added complexity to the measurement estimation without providing any improvement (the state estimation tends to degrade along time) and it required a higher computation time. The minor to major axis ratio has been fixed to 1/3. Then, the length of the major axis is fixed taking into account the distance from the camera to the washbasin, which can be configurable, and the fact that, according to different recommendations, the sleeves have to be rolled up to the elbow. With these states, the discretized system dynamics are given by

$$\begin{bmatrix} x_k \\ y_k \\ \theta_k \\ \dot{x}_k \\ \dot{y}_k \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & T & 0 \\ 0 & 1 & 0 & 0 & T \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_{k-1} \\ y_{k-1} \\ \theta_{k-1} \\ \dot{x}_{k-1} \\ \dot{y}_{k-1} \end{bmatrix} + \mathbf{w}_k \tag{9}$$

where T is the sampling period and $\mathbf{w}_k$ is the noise vector related to the system which determines the spread capability of the particles that identify each hand.

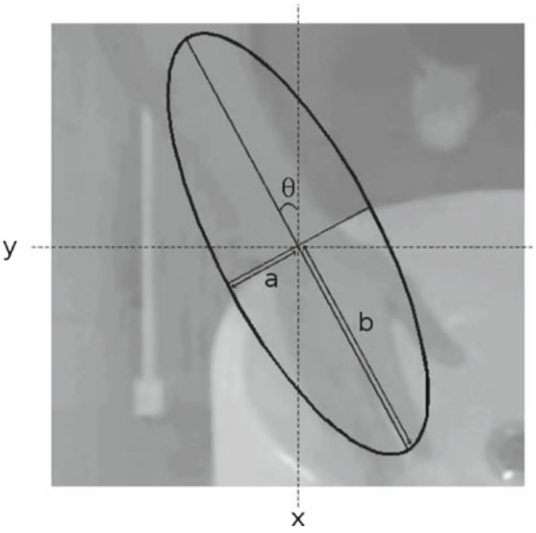

Fig. 6 Parameters of the ellipse model

It is assumed that measurements $\mathbf{y}_k$ are mutually independent and also with respect to the dynamic process. To represent the measurement process, a simple model is used in which Gaussian noise is added. The measurement vector is defined by the ellipse position (x_k, y_k) and orientation (θ_k):

$$\mathbf{y} = \{x_k, y_k, \theta_k\} \tag{10}$$

and the measurement equation is then given by:

$$\begin{bmatrix} x_k \\ y_k \\ \theta_k \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_{k-1} \\ y_{k-1} \\ \theta_{k-1} \\ \dot{x}_{k-1} \\ \dot{y}_{k-1} \end{bmatrix} + \mathbf{v}_k \tag{11}$$

where $\mathbf{v}_k$ is the noise vector related to the accuracy of the measurements.

3 Particle filtering method

The measurements provided by the proposed skin and motion segmentation scheme may contain errors due to several issues such as unfavourable lighting conditions, reflections, changes in the color of the sink surface, occlusions, fast movements of the hands, etc. In addition, in the hand washing scenario the estimation of the position of the hands/arms must be robust over long periods of time and must be able to be re-initialized if the hands are lost, such as when one or both hands leave the scene. These statements support the use of a robust tracking method.

Algorithms using PF to track one or several objects were previously named as *Condensation* [11]. Using independent linear-Gaussian filters to track each one of the objects is not the best solution from the computational point of view. In addition, independent filters may tend to join over the same target, although some techniques have been proposed to resolve this important problem [22]. Kalman filter (KF) is not optimal because it is based on uni-modal Gaussian densities and it is not able to represent multiple alternative hypothesis. Although PF usually are more time consuming, they are also independent on the number of objects being tracked [12,23].

In this application, a uni-modal representation of the system would require to manage a 10 × 1 state vector. Then, the number of particles needed to model all the possible states would be intractable. Accordingly, the estimation for each one of the arms is merged into a single Gaussian multimodal probability density function (pdf) which integrates the state information for each one of the hands/arms. This poses two main challenges. First, we must avoid one of the hands absorbing all the particles of the filter, leaving a uni-modal filter. In a multi-tracking system, there is always one object which yields a higher probability than the others. In our system the number of objects to be tracked is known, and this is

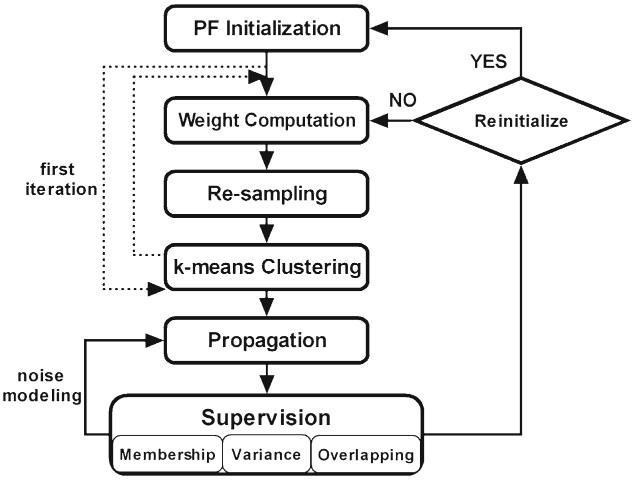

Fig. 7 Outline of the particle filter estimator

used to propagate the particles in an equitable way, assuring the correct representation of the multi-modal pdf. The prior knowledge of the state pdf allows us to force the particles to fill the states we know will represent the system real state. This is performed with a k-means clustering technique which prevents the system from merging in a single Gaussian pdf or to spread in more than two which should represent the two arms. The second challenge is to make the tracking robust to partial occlusions while being able to follow quick movements of the targets. This implies a trade-off solution between a fast response and a stable estimation to partial occlusions. To achieve this, a supervision algorithm which ensures the consistency of the measurements has been developed. This supervision algorithm controls the filter estimations for each of the hands/arms at every filter iteration, and adjusts the filter parameters depending on the occlusion, position and velocity of the arms and the quality of the estimations. This supervision algorithm allows for stable estimations as well as fast filter responses.

The filter can be divided in the following sub-systems (Fig. 7).

3.1 Initialization

In order to cover as much as possible of the probability density function of the state, a fixed number of randomly generated particles are created. At the first filter iteration the particles are assigned to each one of the arms using a clustering algorithm (see Sect. 3.4). The number of particles is equally distributed between the two arms, i.e., once one of the arms has received half of the particles, the rest of the particles are assigned to the other arm. The cluster located on the right side of the image will be labeled as right hand/arm and vice versa. In addition, every time the supervision system detects that the state estimation is not of sufficient quality the filter is restarted and new randomly generated particles take the place of the old ones.

3.2 Weight computation

At this step two tasks are carried out, first, each particle is scored with the probability of being the real state (position, velocity and orientation) of the arms and secondly, an estimation of the state of each of the arms is computed.

Let each particle be represented by its state vector $\mathbf{x}_k$. Then, the probability of representing the real state of the system is computed by using an area-based ellipse fitting method, i.e., for each particle, its weight is computed by summing up the membership (λ) to the skin function of the ellipse inner points:

$$\lambda(\mathbf{x}_k) = \sum_{p_x, p_y \in \text{ellipse}_k} f(p_x, p_y) \tag{12}$$

where $f(p_x, p_y)$ is the probability of point (p_x, p_y) of being a skin pixel, yielded by Eq. (2). This way, as long as a particle/ellipse covers image regions with high probability of being skin it will get high scores. Particles with high scores will have higher probability of being regenerated in the resampling step.

Once the weight is computed for all the particles, the estimation of the real state of the system is also calculated. Different techniques have been explored to find out the one that best represents the real hands/arms position and velocity, while allowing for a fast filter response to fast movements and partial occlusions.

The filter dynamics and observation models have been tuned to rapidly react to changes in the system state. Accordingly, the system's real state is always surrounded by an important number of particles which will quickly absorb fast changes (movements of the arms, occlusions, etc.). These "surrounding particles" will not accurately represent the system's real state but will take care of the fast changes. In fact, these particles will decrease the real state estimation accuracy. To get an accurate estimation of the state, these particles have to be removed from the computation of the estimated state. To do so, the estimated state is computed using an average for an elite of the particles. Using only particles $\mathbf{x}_k$ weighted 80% over the average membership of the particles λ_{mean}, the real state $\hat{x}_i$ for each one of the hands/arms clusters is computed as follows:

$$\hat{x}_i = \frac{1}{M} \sum_{\mathbf{x}_k / \lambda(\mathbf{x}_k) > 0.8 \times \lambda_{\text{mean}}} \mathbf{x}_k \tag{13}$$

where M is the number of particles weighted 80% over the average membership of the particles λ_{mean}. It should be noted that at this stage the particles have already been labeled as belonging to one of the arms, so that Eq. (13) is computed separately for each one of the hands/arms using only its particles.

This, along with the supervision algorithm, allows our system to be stable and accurate in the tracking as well as responsive to changes in the state of the system. When the entire set of particles are used instead of just the elite ones, the estimated state degrades quickly.

3.3 Re-sampling

In the re-sampling step, new particles are randomly regenerated from the old ones depending on the weight/score received on the previous step. Particles receiving a higher score are more likely to be regenerated. To do so, given N particles, a segment [0, 1] is divided into N sub-segments $[na1, na2]$ whose length is proportional to the score achieved by each one of the particles. Then, a random number is generated between 0 and 1. The sub-segment where this number is contained will determine the particle that will be regenerated. Note that the re-sampling process is carried out independently for each arm.

3.4 Clustering

A k-means clustering algorithm is used to split the particles into two clusters (the two arms) after the re-sampling. This information will be used in the next weight computation and re-sampling steps.

The aim of the k-means clustering algorithm is to place a set of vectors in the input space which describes in a discrete way the density of observed samples. To do so it places K random seeds at the first iteration and take them as centroids, linking the samples to their nearest centroid. The samples assigned to each centroid will make a cluster. The centroids are then recalculated as the mean of all of its samples. The algorithm is then repeated until the centroids variation is under a certain threshold. In order to assure that the number of particles is equally distributed between both arms, each cluster can only receive half of the particles, i.e., once one of the arms has received half of the particles, the rest of the particles are assigned to the other arm. The k-means algorithm uses (x, y) particles position in the image as input. The output is a list of labeled particles as 1 or 0. Once the particles have been labeled, the new clusters are linked to the estimation of the arm's position in the last step by using a simple minimum distance criterion.

3.5 Propagation

A dynamic model of the system is used to propagate the particles to the next state. This propagation step is tuned by the supervision algorithm by modifying the variance of the noise added to the dynamic model according to two parameters: degree of overlap between the arms/ellipses (O_a) and the arms/ellipses velocity (v_a). These parameters entail the definition of several thresholds:

$$\begin{cases} \text{if} & O_a \leq \Omega_1 \Rightarrow \text{ separated arms} \\ \text{else if} & O_a > \Omega_1 \text{ and} \\ & O_a \leq \Omega_2 \Rightarrow \text{ joined arms, small overlap} \\ \text{else if} & O_a > \Omega_2 \Rightarrow \text{ joined arms, great overlap} \end{cases} \tag{14}$$

$$\begin{cases} \text{if} & v_a \leq t_v \Rightarrow \text{ slow movements} \\ \text{else if} & v_a > t_v \Rightarrow \text{ fast movements} \end{cases} \tag{15}$$

According to these parameters, the supervision algorithm manages three different situations:

- Separated arms, fast movements: when the last movements have been fast, the system estate is expected to change fast as well. In order to be able to predict these abrupt changes in the state the particles have to be propagated to less expected states. To do so, extra noise is added to the dynamic model so that the predictions adjust better the real movements of the arms. As long as the arms are far away from each other the risk of overlapping is small.
- Joined arms, fast movements: when the area of overlap is small and the movements have been fast no extra noise is added to avoid merging both arms.
- Joined arms, great overlap: when the arms are significantly overlapping since they are very close to each other, a decrease in the noise of the dynamic model is needed in order to avoid the joining of the two arms and also to maintain good estimations to partial occlusions. Low noise levels will make the system state evolution "slow" and so robust to occlusions because the state will be hold longer.

This tuning of the propagation step predicts the evolution of the system as a whole, taking into account not only the last target position and velocity but also the most probable evolution of the system in the next iteration. With this tuning step, the filter is able to follow fast changes in the state as well as to keep good estimates despite partial occlusions.

3.6 Supervision

The supervision algorithm controls the PF state and restarts it when its estimation has degraded. The main reasons for a degradation in the estimations are occlusions, bad initial estimations, and failures in the skin extraction method. The supervision algorithm uses five indicators to evaluate the quality of the filter estimation:

- Membership of the state estimation (one for each arm): For each iteration, the membership of the state estimated by the filter is computed and asked to reach a minimum level. If it does not reach this minimum membership for several consecutive iterations, the supervision algorithm decides that the tracking has lost one or both arms and restarts the filter.

- Variance of the clusters position with respect to the estimated state (one for each arm): For each one of the clusters, the variance of its particles var_i is computed as follows:

$$\mathrm{var}_i = \frac{1}{N} \sum_{\mathbf{x}_k/\mathbf{x}_k \in \mathrm{cluster}_i} (\mathbf{x}_k - \hat{x}_i)^2 \tag{16}$$

  where N is the number of particles of the cluster i, $\mathbf{x}_k$ represents the state of particle k pertaining to cluster i, and $\hat{x}_i$ is the estimated state of cluster i. If the quality of the estimations decreases, then the variance becomes greater because the filter can not set particles around an estimation with high probability. The supervision algorithm checks for this situation for several iterations. If the situation persists, the filter is restarted.
- Overlapping (one for both arms): To prevent the estimations from joining to form a single pdf, the overlap of the solutions for the arm positions is computed on every iteration. If this value is above a threshold the supervision algorithm restarts the filter.

Using this information the supervision algorithm measures the number of objects being tracked, the tracking quality and also is able to determine if both clusters have been assigned to the same arm or not. Finally, the dynamic model is tuned according to the above parameters.

4 Bi-manual gesture descriptors and classification

In this section, we describe the proposed bi-manual gesture recognition scheme. First, the regions of interest that cover both the hands and the arms are obtained. Second, appearance and motion descriptors are computed for the hands and the arms, respectively. Choosing discriminating and independent features is the key to any pattern recognition algorithm to be successful. In this case, the selected descriptors are designed to capture both the single-frame appearance and the relative motion of the different hand washing gestures. Then, a two-staged classifier scheme is applied. Two separate multi-class SVM classifiers are trained from the appearance features and the motion features, respectively. The second stage classifier combines the outputs to produce the final result (see Fig. 8). Finally, once the gestures are determined, a high-level analysis is used to integrate and complete the hand hygiene assessment procedure.

4.1 ROI selection

Once the hands/arms are detected joined, after segmentation and tracking, a ROI selection mechanism is needed to locate the region of interest from which features are going to be extracted. Two ROIs have been defined depending on the feature extraction method. From the appearance point of view, the information related with the arms is not useful, therefore, a ROI covering the hands is selected when appearance features are computed. However, from the motion point of view, the movements of the arms contain relevant information, therefore, a ROI covering the arms is selected when the motion features are used.

The first step consists in computing the intersection point of the two major axes of the ellipses. Then, a vertical symmetry axis is located in the horizontal coordinate of the intersection point. Vertical bounds are first determined and then the horizontal ones, as can be seen in Fig. 9. The width or the height is enlarged to obtain a square region of interest. Finally, the ROIs are resized to 128 × 128 and 192 × 192 for the hands and the arms, respectively. The only difference between both ROIs corresponds to the upper vertical bound, which is fixed to the upper part of the whole image for the case of the ROI that covers the arms. Both processes are depicted in Fig. 9.

4.2 Appearance descriptors

The six different poses clearly differ in shape. The goal is to choose discriminating and independent features to represent

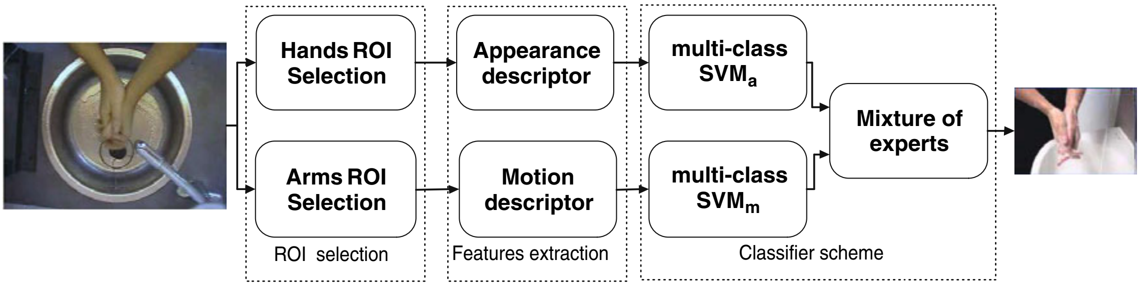

Fig. 8 Global overview of the proposed bi-manual gesture recognition system

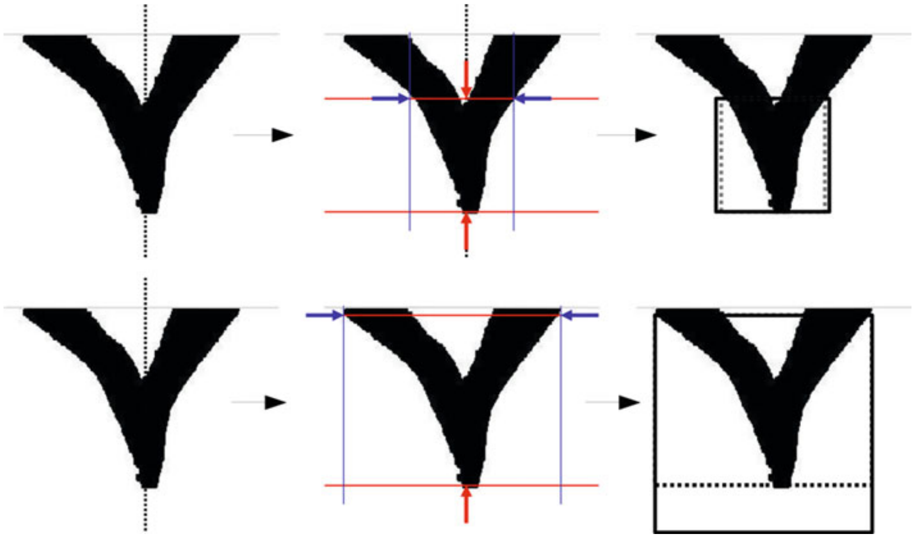

Fig. 9 *Upper row* ROI selection to cover the hands (128 × 128). *Lower row* ROS selection to cover the arms (192 × 192)

the bi-manual gesture shape. We use local histograms of oriented gradients as our single frame feature extraction method, which has reported good performance results in recent applications [6,1]. The aim of this method is to describe an image by a set of local histograms. These histograms count occurrences of gradient orientation in a local part of the image. First, the color image is converted to grey level, then the gradient is calculated, and next the image is split into cells which are defined as square regions with a predefined size in pixels. For each cell, we compute the histogram of gradients by accumulating votes into bins for each orientation. Votes can be weighted by the magnitude of the gradient vector, so that the histogram takes into account the weight of the gradient at a given point.

Due to the variability in the images, it is necessary to normalize the cell histograms. Cell histograms are locally normalized according to values of the neighbored cell histograms. The normalization is done among a group of cells, which is referred to as a block. A normalization factor is then computed over the block and all histograms within this block are normalized according to this normalization factor. We use the L2-norm scheme, $v \rightarrow v/\sqrt{\|v\|_2^2 + \epsilon}$, where v is the unnormalized block descriptor and ϵ is a small regularization constant needed for eventual evaluation of empty gradients. Note that according to how each block has been built, a histogram from a given cell can be involved in several block normalizations. Thus we are going to have some redundant information which, according to [6], improves the performance. Figure 10 shows a graphical description of this method.

When all histograms have been computed for each cell, we build the description vector of an image by concatenating all histograms into a single vector. In order to compute the vector dimension several parameters have to be taken into account: ROI dimension, cell size, block size, number of bins and number of overlapped blocks. Joining together all the elements described so far, the final recount of features is a follows: the ROI size is 128 × 128, each window is divided into cells of size 16 × 16, and each group of 2 × 2 cells is integrated into a block in a sliding fashion, so blocks overlap with each other. Each cell consists of a 16-bin HOG and each block contains a concatenated vector of all its cells. Each block is thus represented by a 64 feature vector which is normalized to an L2 unit length. Thus, in our case we have feature vector dimension of 3.136. Table 1 summarizes the list of all the features included in the final appearance feature vector.

4.3 Motion descriptors

The proposed motion-based bi-manual gesture descriptor is based on optical flow features, induced by the specific motion pattern of each one of the poses. Absolute motions are used since both the camera and the background are static. Spatio-temporal features are introduced to improve the performance of the spatial features at the drawback of requiring temporally aligned training samples. It is expected that the specific dynamics of each one of the poses will be discriminating in the recognition process.

The local orientations of motion edges is captured by emulating the static-image HOG descriptors (see Fig. 11) resulting in the so-called HOF. We use the two optical flow components to find the corresponding flow magnitudes and orientations. These ones are used as weighted votes into local orientation histograms in the same way as for the standard HOG.

In this case, the ROI size is 192 × 192, each window is divided into cells of size 24 × 24, and each group of 2 × 2 cells is integrated into a block in a sliding fashion, so blocks overlap with each other. Each cell consists of a 16-bin

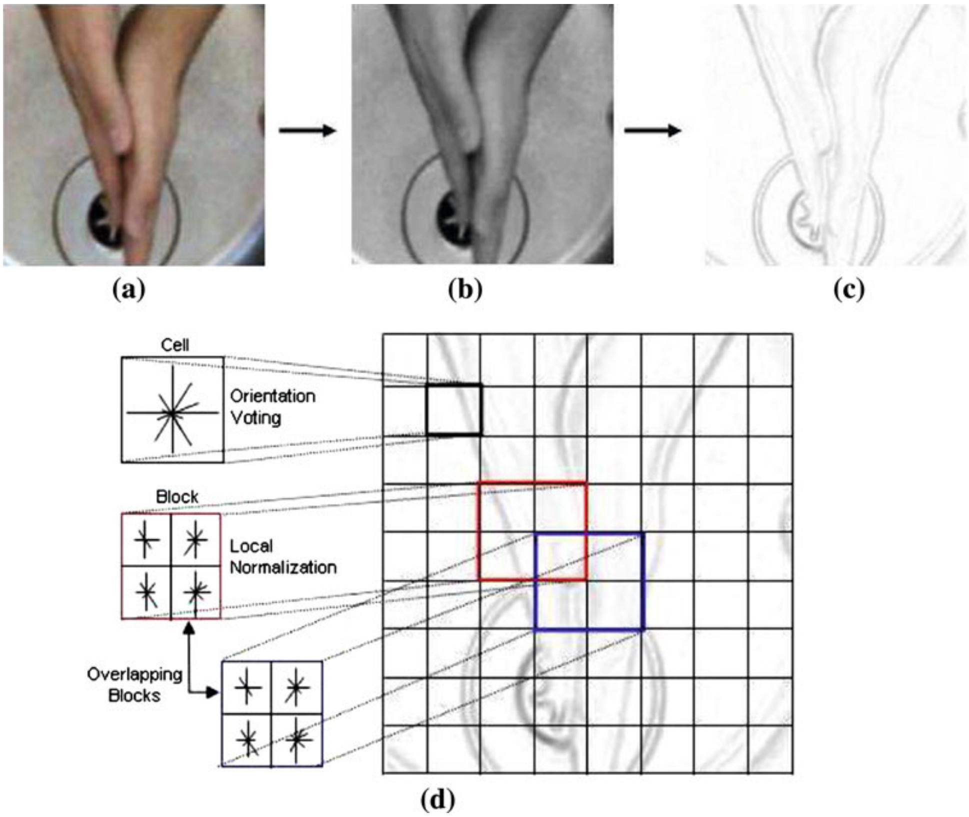

Fig. 10 HOG description. **a** Original image, **b** grey image, **c** gradient image, **d** HOG, cells distribution and blocks normalization

Table 1 List of all the features included in the appearance feature vector

$\text{Cells}_{\text{Row}} = 128/16 = 8$
$\text{CellsBlock}_{\text{Row}} = 32/16 = 2$
$\text{OverlapBlocks}_{\text{Row}} = 8 - 2 + 1 = 7$
$\text{Cells}_{\text{Col}} = 128/16 = 8$
$\text{CellsBlock}_{\text{Col}} = 32/16 = 2$
$\text{OverlapBlocks}_{\text{Column}} = 8 - 2 + 1 = 7$
$\text{VectorDIM} = 7 \times 7 \times 2 \times 2 \times 16 = 3{,}136$

histogram and each block contains a concatenated vector of all its cells. Thus, in this case we have the same feature vector dimension. Table 2 summarizes the list of all the features included in the final motion feature vector.

4.4 Bi-manual gestures classification using multi-class SVM

The SVM classifier is a binary classifier algorithm that looks for an optimal hyperplane as a decision function in a high dimensional space [28]. It is a type of example-based machine learning method for both classification and regression problems. This technique has several characteristics that make it particularly attractive. Traditional training techniques for classifiers, such as multi-layer perceptrons (MLP) use empirical risk minimization and only guarantee minimum error over the training set. In contrast, SVM relies on the structural risk minimization principle [28], which minimizes a bound on the generalization error and therefore should perform better on novel data.

In order to perform the multi-class classification the one-against-one method is proposed, with which $k(k-1)/2$ different binary classifiers are constructed, training data from two different classes for each of them. In this application, there are six different classes relating to the six different poses described in Fig. 1, plus an additional complementary class representing an indeterminate pose (with $k = 7$ we have 21 binary classifiers per each descriptor). In practice the one-against-one method is one of the most suitable strategies for multi-classification problems [10]. After training data from the ith and the jth classes, the following voting strategy is used: if a binary classifier determines that a given sample x is associated to class j, then the vote for the class j is incremented by one. Finally, sample x is predicted to be in the class with the largest vote—this approach is referred to as the *max-wins* strategy [30].

4.5 Mixture of experts

Instead of using a monolithic approach in which appearance and motion features are concatenated into a single feature vector, we propose the use of a mixture of experts architec-

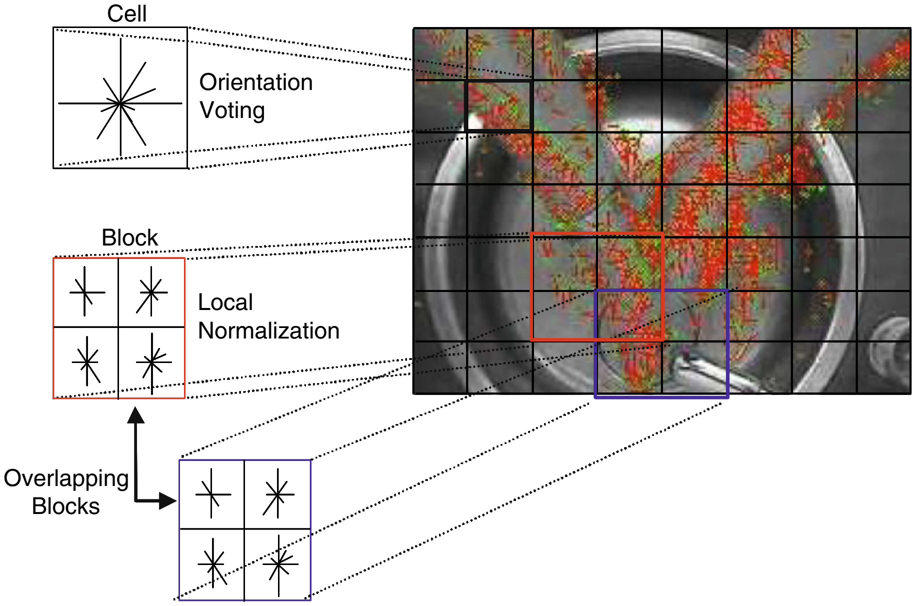

Fig. 11 Motion descriptors, cells distribution and blocks normalization

Table 2 List of all the features included in the motion feature vector

$\text{Cells}_{\text{Row}} = 192/24 = 8$
$\text{CellsBlock}_{\text{Row}} = 48/24 = 2$
$\text{OverlapBlocks}_{\text{Row}} = 8 - 2 + 1 = 7$
$\text{Cells}_{\text{Col}} = 192/24 = 8$
$\text{CellsBlock}_{\text{Col}} = 48/24 = 2$
$\text{OverlapBlocks}_{\text{Column}} = 8 - 2 + 1 = 7$
$\text{VectorDIM} = 7 \times 7 \times 2 \times 2 \times 16 = 3{,}136$

ture. Separate classifiers are independently learned from the appearance and the motion features, and a second stage is used to combine the outputs of these detectors. There are plenty of alternatives in order to define the second stage, for example using a SVM trained to combine the outputs of the first stage. However, we propose to maintain the voting structure using the *max-wins* strategy. As we have 21 binary classifiers per each multi-class classifier, we count 42 votes and determine the class according to the largest vote.

4.6 High-level analysis of the hand washing quality

Poses which are sustained for a threshold number of frames are passed to an accumulator which counts the number of frames that each pose has been maintained for, which effectively counts the amount of time spent in each pose. Each pose has a configurable threshold minimum time (the recommended time for hand washing is a minimum of 15 s of hand rubbing [2,20]). When all thresholds are exceeded the hands are deemed to have been washed to a sufficiently high standard.

The hand pose information is used to complete a hand hygiene assessment. Each of the hand poses must be recorded for a configurable time period. It has to be noted that bi-manual hand washing poses can be executed in any order. This information is integrated over time and provided to the user via a graphical display such as a speaker or an audio display device.

5 Results

The experiments were carried out using standard ceramic and stainless steel sinks and the Kinometrics field test unit (see Fig. 12) constructed for automatic hand wash quality measurement. The system uses the camera sensor to measure the hand wash technique—was it a “splash and go” or did the person follow the hand washing guidelines such as those produced by SARI [24]. It can measure the technique if the person uses soap and water or alcohol gels to clean their hands. This information can be logged as part of HACCP [7] records in a food preparation area.

5.1 Tracking results

In order to support the use of the proposed PF, we compare its performance with the one obtained by a Kalman filter in a set of videos. Each video consists in a sequence with different subjects washing their hands, with different lighting conditions and some of them wearing wristwatches and bracelets. Ground truth data sets have been manually computed using an image graphic tool. Thus, ellipse positions and orientations have been acquired so that performance comparisons can be made between both filters and the ground truth data.

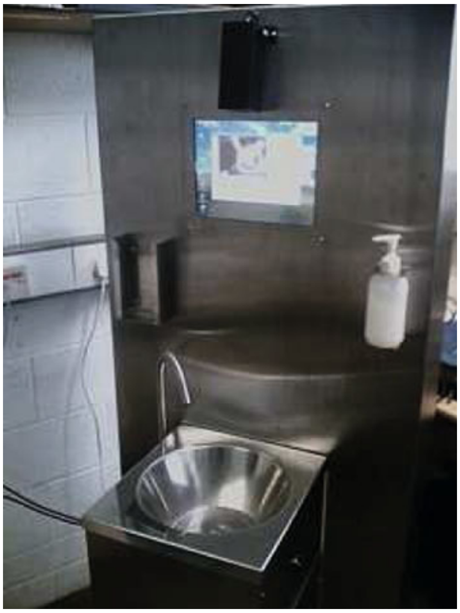

Fig. 12 Field test unit. The hand wash station uses the camera (*black object*) to measure the quality of the hand wash and results are displayed on the touch screen

To evaluate and compare the filters performance the root mean square error (RMSE) is computed:

$$\mathrm{RMSE}(\hat{x}) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x(i) - \hat{x}(i))^2} \tag{17}$$

Input images for both filters are the ones yielded by the skin and motion segmentation algorithms described in Sect. 2. State and measurement noise matrices were chosen to be the same either for the PF or the KF. In the case of the PF, several experiments were carried out with different number of particles. An area-based ellipse fitting method has been also applied to get the measures for the KF estimator. The prediction yielded by the KF is used to restrict the region where the next measures are taken. A threshold operator combined with morphological filters is applied to the skin probability images to reduce the number of points where the ellipses are centered and thus, reduce the number of operations needed by the area based ellipse fitting process. The measures yielded by this method are very similar to the measures taken by the PF estimator (Table 3).

As stated in Sect. 2.2, the major axis of the ellipses is fixed by configuring the distance between the camera and the washbasin (this parameter is fixed in the Kinometrics field test unit) and by taking into account that the user should have the sleeves rolled up to the elbow. Then, the minor axis is defined as one third part of the major one. With this assumption good results are achieved at a reasonable computational cost. On the contrary, if we include both or one of the ellipse axis in the model, the computational time required to achieve a solution becomes intractable and the state estimation tends to degrade with time.

Table 3 Effect of the number of particles on the performance

# Particles	RMSE orientation	RMSE position
$N = 50$	8.2956	12.9675
$N = 75$	8.2736	10.6263
$N = 100$	7.6524	9.8123
$N = 150$	7.8920	7.9388
$N = 200$	7.5046	8.5176
$N = 250$	7.3416	8.1302
$N = 300$	7.1035	7.5869
Kalman	25.3185	12.6010

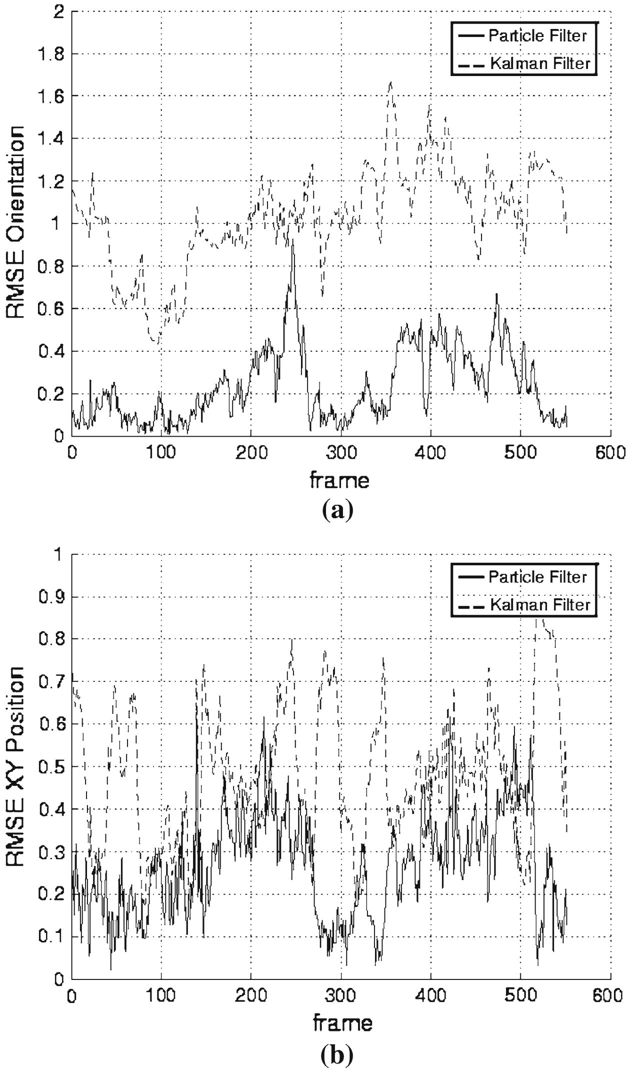

Fig. 13 RMSE analysis for Particle and Kalman filters. **a** Orientation θ. **b** XY Position

Regarding both an accurate response and a reasonable computation time, the ideal number of particles is 200.

Table 3 depicts RMSE for position and orientation for PF with different number of particles and for KF. It is shown that PF yields better results for orientation as well as for position estimations. As long as the number of particles increases, the RMSE decreases and less time is needed to get stable estimations. On the other hand, the computation time increases with the number of particles. The higher performance of the PF can be explained by the multi-modal and non-linear nature of the problem to estimate. KF is able to predict the state of the system as long as it remains "inside the limits" of linearity. When both arms overlap or move too fast the KF fails in its predictions.

Figure 13 depicts the RMSE along time for the PF an KF. As can be seen PF yields better estimations for arms orientation and position. Also PF is robust to head occlusions and hands overlapping. It is able to maintain the estimation of the pose and the orientation for a few frames thanks to the supervision algorithm which adjust the filter response to the estimated situation of the system. It also performs better to quick movements of the arms, where the KF shows some inertia and sometimes loses the arms. The supervision algorithm restarts the estimation when occlusions or overlapping areas persist in time. Figure 14 provides two examples where the results of the PF tracker can be seen.

5.2 Classification results

We tested our system using a database of hand poses which was built up in the following way: we recorded 6 videos about 600 frames each one (24 s long) in a single session. Each video consisted of a sequence of hands performing the movements associated to only one specific pose (four different subjects per video). Pose 7 appears in all the sequences. The subjects were instructed to wash their hands following the guidelines suggested by [20]. In this case, a stainless steel hand wash basin has been used under unfavorable lighting conditions. Thus, we can assure that the classifier has to deal with the worst case scenario: specular reflections and low contrast between the hands and the sink.

Next, the experts labeled each frame as pose 1 to 6, using the label 7 for the indeterminate pose, and those frames in which the experts did not agree were rejected. Following, the selected frames were analyzed by the ROI segmentation procedure and all those frames in which the ROI could not be detected by the system because the hands were not joined—basically at the start and at the end of the video sequences—were rejected. The remaining frames—about a half of the original data set—constitute the training set with which the SVM ensemble was trained. The test set consisted of two videos of about 1 min long in which the different poses were randomly visualized. The three experts performed a labeling of all the frames for these two videos, and all those frames in which the experts did not agree were rejected. In addition, all the frames in which a ROI was not segmented by the system were not taken into account. The remaining frames (1,449/1,924 for video 1 and 1,025/1,309 for video 2) constitute the test set (see Table 4).

Segmented ROIs, from where features were extracted, were resized to a fixed size of 128 × 128 or 192 × 192 (depending on the features) to have a normalized area of analysis. For the motion analysis we applied the Lucas and Kanade approach [17] with a threshold value $T_{ID} = 50$ and a window of 7 pixels for the average filter. We used the LIBSVM library [5] for building up the SVM classifier ensemble

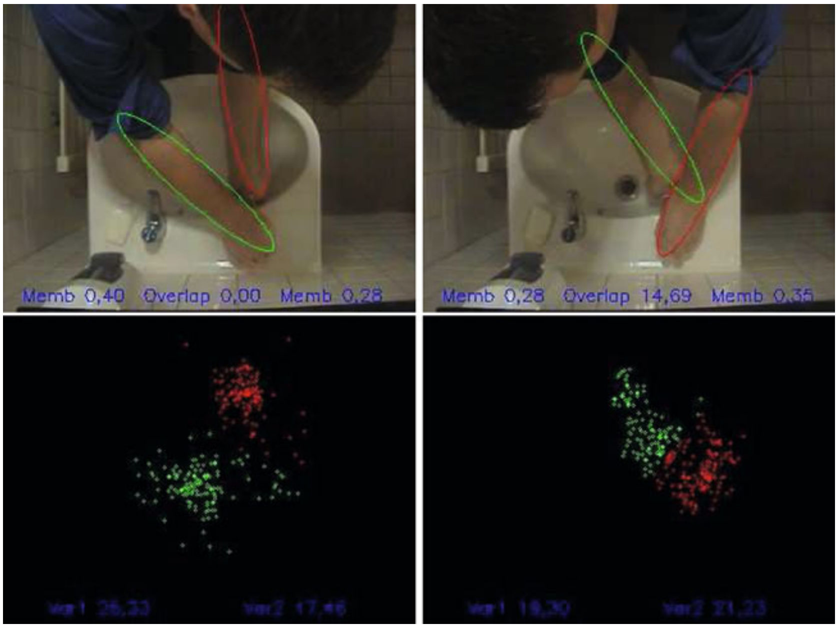
Memb 0,40 Overlap 0,00 Memb 0,28
Memb 0,28 Overlap 14,69 Memb 0,35
var1 26,33 var2 17,46
var1 19,30 var2 21,23

Fig. 14 Tracking sequence. *Upper row* Particle filter results. *Lower row* $X-Y$ particles distribution

Table 4 Training and test data sets

Data sets	Pose 1	Pose 2	Pose 3	Pose 4	Pose 5	Pose 6	Others
Training	638	780	594	977	902	1,073	587
Test	488	414	360	456	477	358	304

Table 5 Detection rates (DR) for the different classes

Descriptors	Pose 1 (%)	Pose 2 (%)	Pose 3 (%)	Pose 4 (%)	Pose 5 (%)	Pose 6 (%)	Others
HOG DR	86.07	91.55	94.72	89.25	86.37	96.09	61.84
HOF DR	87.76	96.63	85.00	82.71	65.99	96.78	71.93
MoE DR	88.32	97.81	94.71	89.31	85.95	96.13	70.66
Inter-observer agr.	94.59	97.71	97.20	97.53	98.46	97.43	82.98

Table 6 Distribution table of the multi-class ensemble

Poses	System						
	1	2	3	4	5	6	7
Human							
1	241	0	2	7	0	0	4
2	0	201	0	0	11	0	0
3	0	2	170	0	3	0	0
4	1	0	0	294	1	3	0
5	3	0	0	5	193	11	4
6	3	0	0	0	0	143	2
7	1	4	0	5	17	5	113

with radial basis function (RBF) kernels. All the single classifiers used a $\gamma = 3.2e^{-04}$ as the RBF parameter, which was obtained empirically. We can see the results after applying multi-classification with HOG and motion features and SVM classifier in Table 5.

On the one hand, appearance features perform better for poses 3, 4 and 5. On the other hand, motion features provide better results for poses 1, 2 and 7. Pose 6 is detected with almost the same accuracy in both cases. The mixture of experts architecture combines the output of the two detectors and performs slightly better than the best monolithic classifier. Although detection rate is low for the "other poses" case, which is reasonable taking into account that it is the class with highest variability, all the recommended poses are classified with detection rates above 85%. Three of them are classified with an accuracy greater than 94%, and the best classified class has a detection rate of 97.81%. The last row of Table 5 shows the inter-observer agreement as the percentage of coincident labels assigned by different experts to frames of the same class, which provides information about the intrinsic difficulty of each pose. The inter-observer agreement was calculated in the following way: we take all the M^1 frames labeled as class 1 by Expert 1 and Expert 2 and we count the number of coincidences $N_1^{1,2}$. The inter-observer agreement for Experts 1 and 2 in class 1 is calculated as the ratio $N_1^{1,2}/M^1$. We repeat the same procedure for Experts 2 and 3, and for Experts 1 and 3, and we average the results (this same scheme is applied for the rest of the classes).

The results shown in Table 5 appear to confirm that the proposed approach behaves in a similar way as the experts: those classes which show the highest detection rate, as in the case of Pose 2 and Pose 6, tend to show a high expert agreement, and vice versa, both low agreement and detection rate in Pose 7, perhaps with an exception in the case of Pose 5. This fact can be analyzed in the distribution table of Table 6, in which the element in the row i and column j represents the number of frames labeled by the human as Pose i which were classified by the system as Pose j. The distribution matrix shows how most of the misclassified samples of Pose 5 are voted for Pose 6 and the "other poses" class. However, the system appears to achieve a better generalization of Pose 6, since no misclassified frames are voted for Pose 5 in this case.

Figure 15 depicts an example of the single frame classification results in a given sequence. The detected class is shown with a number at the top left of each image. The system correctly detects when the hands are separate or joined. Even with a single frame approach the classifier correctly detects the transitions between poses, which are classified as the "other poses" class (a video sample of the system output can be obtained via web in [13]).

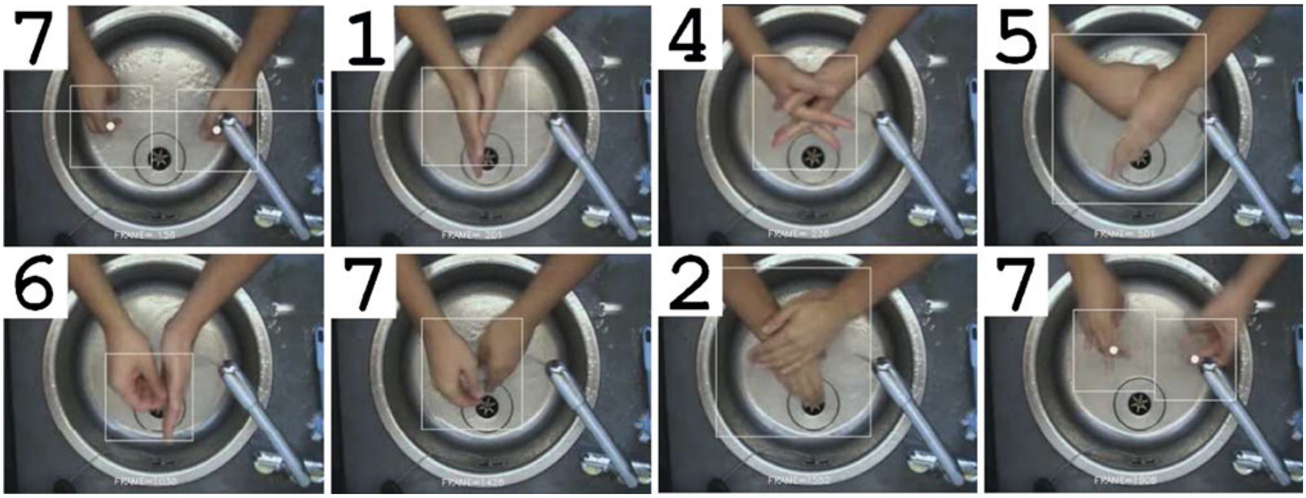

Fig. 15 A sequence of frames with their associated pose class

6 Conclusions and future work

In this work, we present a vision-based system for automatic hand washing quality measurement which can be used in hospitals, clinics, food processing industry, etc. One camera is placed over the sink focusing on the hand washing area. The system is implemented in a PC-based architecture and the complete algorithm runs at an average rate of 20 frames/s. Hands/arms are segmented by means of skin and motion features. This segmentation process is robust against specular reflections due to steel sinks, illumination changes, etc.

A robust estimator of hands/arms position, orientation and velocity based on a probabilistic multi-modal filter, completed with a k-means clustering technique, is proposed. The RMSE analysis has been used to describe and compare performance. The obtained results showed that the PF estimator performs better than the usual KF estimator. It is robust to partial occlusions and fast movements whereas KF is only able to track soft movements.

Two independent SVM classifiers are learned from appearance (HOG) and motion (HOF) features. Each classifier contains 21 binary classifiers (one-against-one) and their outputs are provided using the *max-wins* strategy. A mixture of experts architecture is used in a second stage, combining the outputs of the monolithic detectors. This classification structure appears to yield more than acceptable results. The introduction of spatio-temporal features improves the overall performance of the system. Symmetric hand washing poses (which can be either done with left or right hand) are correctly detected. The worst detected class is the one with highest variability (*other poses*) which has a detection rate of 70.66%. The minimum and maximum detection rate for the six poses are 85.95% (Pose 5) and 97.81% (Pose 2).

As a novel application, the outcomes shown in this contribution are encouraging results for a multi-class classification problem with seven classes. However, several tasks are planned for the future in order to both improve the performance and reduce the computation costs. The search of an optimal trade-off between the discrimination power of the feature vector and its size is an important issue in order to reduce the overall computation load in on-line systems. Single frame classification can be improved by adding a multi-frame validation process, in which stochastic models (hidden Markov models, for instance) might play an important role integrating sequence information. New data sets will be created for training and test, under various lighting conditions, with different types of sinks and a larger variety of subjects, in an attempt to improve the robustness and the generalization capability of the proposed system. Finally, although the multi-class SVM ensemble appears to yield good classification results, the use of other alternatives with a lower computational cost are in our scope, looking forward to integrate this application into a feasible low cost FPGA platform in order to successfully deal with the development of low cost wireless sensors to measure human activity [13]. All these issues are part of our current lines of research, and will be studied in detail in future contributions.

Acknowledgments This work has been partially supported by Enterprise Ireland Project CFTD/2006/216.

References

1. Alonso, I.P., Llorca, D.F., Sotelo, M.A., Bergasa, L.M., Revenga, P., Nuevo, J., Ocaña, M., Garrido, M.A.: Combination of feature extraction methods for SVM pedestrian detection. IEEE Trans. Intell. Transport. Syst. **8**(2), 292–397 (2007)

2. Boyce, J.M., Pittet, D.: Guideline for hand hygiene in health-care settings. Morb. Mortal. Wkly. Rep. **51**(RR-16), 1–45 (2002)
3. Bradsky, G.R., Davis, J.W.: Motion segmentation and pose recognition with motion history gradients. Mach. Vis. Appl. **13**, 174–178 (2002)
4. Chen, P., Grecos, C.: A fast skin region detector for colour images. In: Proceedings of IEEE International Conference on Visual Information Engineering (2005)
5. Chang, C.C., Lin, C.J.: LIBSVM: A library for support vector machines. http://www.csie.ntu.edu.tw/~cjlin/libsvm (2001)
6. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (2005)
7. HACCP: A Food Safety Management System based on the Principles of Hazard Analysis and Critical Control Point, Food Safety. Authority of Ireland (2006)
8. Hoey, J.: Tracking using flocks of features, with application to assisted handwashing. In: Proceedings British Machine Vision Conference (2006)
9. Hoey, J., von Bertoldi, A., Poupart, P., Mihailidis, A.: Assisting persons with dementia during handwashing using a partially observable Markov decision process. In: Proceedings of International Conference on Vision Systems (2007)
10. Hsu, C.W., Lin, C.J.: A comparison of methods for multi-class support vector machines. IEEE Trans. Neural Netw. **13**(2), 415–425 (2005)
11. Isard, M., Blake, A.: CONDENSATION—conditional density propagation for visual tracking. Int. J. Comput. Vis. **29**(1), 5–28 (1998)
12. Isard, M., McCormick, J.: BraMBLe: A Bayesian multiple-blob tracker. In: Proceedings of IEEE International Conference on Computer Vision (2001)
13. Kinometrics: Low cost wireless sensors to measure human activity. http://www.kinometrics.com (2007)
14. Lee, H.-J., Chung, J.-H.: Hand gesture recognition using orientation histograms. In: Proceedings of IEEE Region 10 Conference TENCON (1999)
15. Lin, D.-T.: Spatio-temporal hand gesture recognition using neural networks. In: Proceedings of IEEE World Congress on Computational Intelligence (1998)
16. Llorca, D.F., Vilarino, F., Zhou, J., Lacey, G.: A multi-class SVM classifier for automatic hand washing quality assessment. In: Proceedings of British Machine Vision Conference (2007)
17. Lucas, B.D., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: Proceedings of IUW (1981)
18. Marcel, S., Bernier, O., Viallet, J.-E., Collobert, D.: Hand gesture recognition using input–output hidden Markov models. In: Proceedings of IEEE International Conference on Automatic Face and Gesture Recognition (2000)
19. Kolsch, M., Turk, M.: Robust Hand Detection, In. Proceedings of IEEE International Conference on Automatic Face and Gesture Recognition (2004)
20. MRSA: Methicillin-resistant *Staphylococcus aureus*. Guidance for nursing staff, Royal College of Nursing (2005)
21. Pavlovic, V., Sharma, R., Huang, T.: Visual interpretation of hand gestures for human–computer interaction: a review. IEEE Trans. Pattern Anal. Mach. Intell. **19**(5), 677–695 (1997)
22. Qu, W., Schonfeld, D., Mohamed, M.: Real-time distributed multi-object tracking using multiple interactive trackers and a magnetic-inertia potential model. IEEE Trans. Multimedia **5**(3), 511–519 (2007)
23. Sanjeev, M., Maskell, S., Gordon, N., Clapp, T.: A tutorial on particle filters for online nonlinear/Non-Gaussian Bayesian tracking. IEEE Trans. Signal Process. **50**(2), 174–188 (2002)
24. SARI: Guidelines for hand hygiene in Irish Health Care settings. A strategy for the control of antimicrobial resistance. Infection Control Subcommittee (2004)
25. Starner, T., Pentland, A.: Visual recognition of American sign language using hidden Markov models. In: Proceedings of International Workshop on Automatic Face and Gesture Recognition (1995)
26. Storring, M.: Computer Vision and Human Skin Color. Faculty of Engineering and Sciencem Aalborg University, Niels Jernes Vej 14, 9220 Aalborg, Denmark (2004)
27. Su, M.-C., Huang, H., Lin, C.-H., Huang, C.-L., Lin, C.-D.: Application of neural networks in spatio-temporal hand gesture recognition. In: Proceedings of IEEE World Congress on Computational Intelligence (1998)
28. Boser, B.E., Guyon, I.M, Vapnik, V.N.: A training algorithm for optimal margin classifiers. In: ACM workshop on COLT, pp. 144–152 (1992)
29. Viola, P., Jones, M.: Robust real-time object detection. Int. J. Comput. Vis. **57**(2), 137–154 (2002)
30. Wu, T.F., Lin, C.J., Weng, R.C.: Probability estimates for multi-class classification by pairwise coupling. J. Mach. Learn. Res. **5**, 975–1005 (2004)