A Machine Vision-Based Gestural Interface for People With Upper Extremity Physical Impairments

Hairong Jiang, Bradley S. Duerstock, and Juan P. Wachs, Member, IEEE

Abstract—A machine vision-based gestural interface was developed to provide individuals with upper extremity physical impairments an alternative way to perform laboratory tasks that require physical manipulation of components. A color and depth based 3-D particle filter framework was constructed with unique descriptive features for face and hands representation. This framework was integrated into an interaction model utilizing spatial and motion information to deal efficiently with occlusions and its negative effects. More specifically, the suggested method proposed solves the false merging and false labeling problems characteristic in tracking through occlusion. The same feature encoding technique was subsequently used to detect, track, and recognize users’ hands. Experimental results demonstrated that the proposed approach was superior to other state-of-the-art tracking algorithms when interaction was present (97.52% accuracy). For gesture encoding, dynamic motion models were created employing the dynamic time warping method. The gestures were classified using a conditional density propagation-based trajectory recognition method. The hand trajectories were classified into different classes (commands) with a recognition accuracy of 95.9%. In addition, the new approach was validated with the “one shot learning” paradigm with comparable results to those reported in 2012. In a validation experiment, the gestures were used to control a mobile service robot and a robotic arm in a laboratory chemistry experiment. Effectiveness control policies were selected to achieve optimal performance for the presented gestural control system through comparison of task completion time between different control modes.

I. INTRODUCTION

ASSISTIVE technologies is about finding new ways to engage cutting-edge technologies in support of individuals with physical and/or cognitive impairments. The development of technologies relying on high usability principles, exploited new communicational channels, such as eye blinking, voice, hand gestures, sip and puff, and electromyogram (EMG) as effective control modalities [1]. These channels have led to ingenious interfaces in support of the disabled [2], such as intelligent wheelchairs systems, home medical alert systems, and assistive robotic control, to mention a few [3]–[5]. These interfaces offer additional degrees of mobility and control which were not possible prior to these developments, leading to a higher quality of life and sense of independence.

Among all these interaction channels, hand gestures is a valuable alternative since it does not require the user to be tethered through cables or sensors, and it only requires learning a few customized gestures for a given task. In particular, upper extremity gesture control can serve as an important human computer interaction (HCI) modality for individuals with quadriplegia who lack hand fine motor skills. For instance, upper limb gesture control requires less targeting accuracy than joysticks, the mouse, and other continuous input devices. Likewise, the option to employ either continuous or discrete input control modes reduces the effort required for individuals with quadriplegia to perform navigational operations [6]. Unlike voice control, gesture control is effective in noisy environments [7]. In addition, for most of the cases, individuals with quadriplegia can only use gross motor instead of fine motor function to perform certain tasks [8]. Apart from other common modalities, such as keyboard and joystick that require fine motor control to hit a key or move and twist a handle, upper extremity gesture control only requires gross motor function for targeting and navigational tasks [9]. Lastly, hand gesture-based HCI is unencumbered because it does not require the user to directly contact or wear sensors as sip-and-puff and EMG-based systems [10], [11]. While not every individual with upper extremity mobility impairments can use hand gesture control reliably, for those who are able to move their arms to some degree, gesture-based HCI can be seen as a promising alternative or complement to an existing control modality.

In our previous work [12], a prototype of gesture recognition-based interface was developed for people with upper extremity mobility impairments. In the current manuscript, the tracking algorithm was greatly improved and compared with five state-of-the-art algorithms to demonstrate a better tracking performance. Further, more experimental results were provided with subjects with upper extremity mobility

Authorized licensed use limited to: Purdue University. Downloaded on June 25,2020 at 21:57:13 UTC from IEEE Xplore. Restrictions apply.
impairments and one shot learning was employed to allow instant customization of the gestural system. Face and hand tracking under frequent self-occlusion was modeled as a multiobject tracking (MOT) problem. This problem is challenging since hands are nonrigid objects and their form varies among individuals, while performing a certain candidate gesture. Additionally, since the appearance of the left and right hand are similar for the same individual, trackers can focus on one hand or exchange positions when the hands are too close to each other. In this paper, an integrated approach was proposed to tackle the challenging problem of tracking under self-occlusion.

A Related Work

Often, hand gesture recognition involves segmentation of the hands, tracking them through occlusion, and the classification of hand’s dynamic trajectories and static pose. For vision-based real-time gesture-based interfaces for assistive technologies, robustness is a critical requirement [13] for its adoption. For hand segmentation, a commonly used method is to back-project the prebuilt skin color histogram model into new video frames. These methods are likely to fail in true world conditions, where illumination is uncontrolled and the background is cluttered. Adding depth information can relax at some extent the problem, by utilizing stereo vision [14] or other depth commodity sensors, such as Kinect [15] or Leap Motion [16].

Face and hands tracking is a special case of MOT problem. If gestures in the lexicon only carry trajectory information, (the hand shape does not convey extra information), classical tracking approaches can be adopted. For example, CAMSHIFT [17] and conditional density propagation (CONDENSATION) [18] have been shown to successfully track the hands; however, they are susceptible to lose the tracked hands when occluded by new objects, or when the scene illumination changes. Another widely used technique for object tracking is particle filters [19]. Perez et al. [20] integrated color-based appearance models to a particle filter framework to enhance tracking under complex background and occlusion, and then applied the particle filter framework to multiple objects tracking. Okuma et al. [21] further extended particle filters by incorporating a boosting detector and enabling automatic initialization of potential multiple targets. One problem of these techniques is that the interaction between the tracked objects (and occlusion) was not considered part of the main framework. When the objects interact one with the other, occlusions occur frequently. Local motion information was incorporated into a color-based particle filter framework by Kristan et al. [22] to solve the self-occlusion problem through object tracking. Qu et al. [23] combined a joint state space representation with color-based particle filter and performed joint data association in a multi-object tracking scenario. All the discussed algorithms so far, attempted to solve the MOT problem; however, they presented limited performance when tracking multiple nonrigid similar objects.

With the advent of Kinect and other 3-D sensors, hand or body tracking techniques in real-time were exploited. Eichner et al. [24] presented a technique to estimate the body layout of humans by using still images. Their approach is capable of estimating upper body pose in highly uncontrolled environment. Further, Yang et al. [25] described a method to estimating human pose from static images using body part models. By using the depth information, Shotton et al. [26] proposed a method to predict 3-D positions of body joints from a single depth image. They solved the pose estimation problem through a simple per-pixel classification problem. A similar method is also used by OpenNI for human body skeleton tracking. One problem of these skeleton-based tracking methods is that they work well when users are standing with their extremities extended, but suffer performance degradation for seated users with contracted limbs, which normally occurs with individuals with quadriplegia.

Only color and depth information captured from Kinect were adopted for hand tracking by Okonikomis et al. [27]. They presented a method to track the full articulation of two hands that interact with each other in an uncontrolled manner. This method is effective for static gesture recognition; however, the computation cost is excessive which affects its real-time extension for gesture tracking. One of the most widely used techniques for gesture recognition is Hidden Markov Models (HMM) [28]–[30]. Common problems with HMM approach consist of finding the optimal parameters set (e.g. initial probabilities) and trajectory spotting for gesture temporal segmentation. Black and Jepson [31] proposed a CONDENSATION-based trajectory gesture recognition algorithm that can obtain less sensitive parameters set and achieve robust tracking, yet gesture temporal segmentation was not fully addressed. Akam, et al. [32] applied the dynamic time warping (DTW) approach to gesture recognition and look at subgestures composition to solve the temporal segmentation problem (also known as spotting). Interaction between hands was not specifically tackled.

Recently, a new type of challenge was attracted the attention of the gesture recognition community – the One Shot Learning Challenge [33]. The one shot learning [34] consists of learning a gesture category by only observing one instance of that gesture, similar to how humans learn. In this context, Wu et al. [35] adopted the extended-motion-histogram image for motion feature representation and applied it to segment and classify hand gestures. Yang et al. [36] proposed discovering high level subactions by clustering optical flow in four dimensions (RGB-D). In our work, one shot learning provides an interesting test-bed to demonstrate the robustness of our approach, compared with the state of the art [37]. In the current paper, we also extended our method to robotic control. One advantage of using hand gestures to control robots is that it provides a natural way for navigational tasks by sending navigational information (e.g., left, right, forward, and backward commands) [38].

B. Outline of Our Approach

In this paper, an interaction model was incorporated to the color histogram-based particle filter framework to track hands through interaction and occlusion. A procedure was proposed to create dynamic motion models by DTW method and classify input gesture trajectories using the CONDENSATION
Algorithm. The system was integrated in a simplistic yet robust fashion by combining CONDENSATION algorithm with an interaction model-based particle filter, which makes it suitable for human robot interaction in assistive technologies. The contribution of this paper is three-fold:
1) prove the effectiveness of hand gestures as an alternative modality for individuals with mobility impairments by both subjective explanation and quantified results;
2) solve the frequently hand gesture interaction and occlusion problem through integration of color and 3-D spatial information as an interaction model;
3) new gestures can be created and learned through the one shot learning paradigm, leading to an almost effortless training process (a necessary attribute for subjects with severe spinal cord injuries).

The paper is organized as follows. In Section II, the architecture is presented for the gesture recognition system. In Section III, the approach suggested to track and recognize dynamic hand gestures is discussed in details. In Section IV, comparative tests and results are presented, Section V discusses and concludes the paper, and Section VI presents future work.

II. SYSTEM ARCHITECTURE

The architecture of the proposed system is illustrated in Fig. 1. Eight gestures were selected to constitute the gesture lexicon which in turn was used to control the robots. The machine vision-based gestural system included four parts: foreground segmentation, face and hand detection and tracking, hand trajectory classification, and robotic control policies. These parts are described in the following sections.

A. Foreground Segmentation

In foreground segmentation section, the background was ruled out from the captured frames and the whole human body was kept as the foreground.

B. Face and Hand Detection and Tracking

Face and hand detection was used to initialize the position of the face and hands for the tracking phase. After initialization, both face and hands were tracked through video sequences by the particle filter method.

C. Hand Trajectory Classification

Hand tracking results were segmented as trajectories, compared with motion models, and decoded as commands for robotic control.

D. Robotic Control Policies

The commands decoded by gesture recognition results were sent to control the mobile robot and the robotic arm.

III. GESTURE RECOGNITION

A. Foreground Segmentation

Initially, the user’s body was treated as a foreground object in order to detect the user’s movements. Two steps were used to segment the foreground (refer to algorithm 1 in Table I). In the first step, the sensed image acquired by a Kinect [15] sensor was thresholded using depth information. The depth value of each pixel was denoted as \( D(i, j) \) with \( i \) and \( j \) indicating the horizontal and vertical coordinates of the pixel in each frame of the video sequence. An example of a depth image is shown by Fig. 2(a), where the distance between objects and the depth sensor was mapped to intensity levels. The nearer the object was to the sensor, the larger the intensity was. Two absolute depth thresholds (a low threshold \( T_{DL} \) and a high threshold \( T_{DH} \)) were custom set by the user according to their relative distance to the depth sensor.

\( T_{DL} \) was set to no less than a constant which was the minimum distance that can be registered by the depth sensor (due to its physical limitations). \( T_{DH} \) was set to be the maximum distance that can be reached by the user while seated in a wheelchair. These values are selected since they resulted in the best performance; other thresholds can be used and the impact on the overall performance is likely to be negligible.

Authorized licensed use limited to: Purdue University. Downloaded on June 25,2020 at 21:57:13 UTC from IEEE Xplore. Restrictions apply.
at time \( t \),

\[
p(Z_t | X_t) \propto \frac{1}{\sqrt{2\pi} \sigma_t} \exp\left(-\frac{1}{2\sigma_t^2}(X_t - Z_t)^2\right)
\]

where \( \lambda_1 \) measures the variance of the HSV histogram. Equation (5) can be rewritten by adding a normalization factor \( k \) to normalize the sum of all particles’ weight to one, obtaining

\[
p(Z_t | X_t) = k \exp(-\lambda_1(D)^2).
\]
function for MC model is defined as

\[ \psi_1(X_{i,t}, X_{j,t}) = \frac{1}{\lambda} \exp \left( -d(X_{i,t} - X_{j,t})^2 \right) \]

The likelihood function for the MC model is defined as

\[ \psi_r(X_{i,t}, X_{j,t}) = \frac{1}{\lambda} \exp \left( -d(X_{i,t} - X_{j,t})^2 \right) \]

For CP model, as in [43], we have

\[ p(X_{i,t}|X_{i,t-1}) \propto \prod_{j \neq i} \psi_1(X_{i,t}, X_{j,t}) \prod_{j \neq i} \psi_r(X_{i,t}, X_{j,t}) \]

The particle filter function (2) can be rewritten as

\[ p(X_{i,t}|Z_{i,t}) = k \prod_{j \neq i} \psi_1(X_{i,t}, X_{j,t}) \prod_{j \neq i} \psi_r(X_{i,t}, X_{j,t}) \]

D. Hand Tracking Through Interaction and Occlusion

Color-based particle filter tracking was effective for multiple independent objects tracking when the objects did not interact or occlude each other. However, if interaction or occlusion occurs, multiple independent particle filters can be used. Standard MOT with interaction and occlusion suffers from the false merging and false labeling problems [23]. The false merging problem denotes the situation that the tracker shift from the object being tracked to a different object that has higher observation likelihood. Conversely, the false labeling problem denotes the situation that the objects being tracked exchange their labels after interaction or occlusion occurred.

In the proposed system, the face and both hands were tracked. When the particles interact or occlude each other, however, if interaction or occlusion occurs, multiple independent trackers were applied to the objects (Fig. 4(a)). However, when the objects’ observations interact (e.g. partial or complete occlusion occurs), the conventional particle filter framework is extended (Fig. 4(b) and (c)). The decision of when the objects interact is made based on the distance between the hands. When this distance is below a decision threshold, the system switches to the interaction model (Fig. 4). To find the optimal threshold, a histogram of the number of tracking frame errors at each distance is obtained (Fig. 5) and the threshold is selected so the tracking errors are minimized when the interaction model is activated. Note, a dramatic decrease of error, at the distance around 50 pixels, at which hand interaction frequently occurred. The threshold \( T \) was determined according to the distribution of errors in

Authorized licensed use limited to: Purdue University. Downloaded on June 25,2020 at 21:57:13 UTC from IEEE Xplore. Restrictions apply.
JIANG et al.: MACHINE VISION-BASED GESTURAL INTERFACE FOR PEOPLE WITH UPPER EXTREMITY PHYSICAL IMPAIRMENTS

Fig. 4. Dynamic motion analysis. (a) Independent object tracking. (b) Interaction model added. (c) Objects occlusion occurs.

Fig. 5. Number of error tracking frames for each distance before and after the addition of interaction model.

the histogram. The extension models were given through (8) and (9). The parameters \( \lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5, \) and \( \lambda_6 \) in (6), (8), and (9) were optimized by utilizing a neighborhood search method [38]. The algorithm for hand tracking during interaction and occlusion is shown in Table II.

E. Gesture Lexicon
A gesture lexicon was designed such that users will physical impairments can perform the gestures with minimal effort. These gestures were found through a series of interviews conducted with subjects with upper mobility impairments. Borg Scale [44] was used to rank the physical stress required to perform a gesture by participants with upper mobility impairments. An eight-gesture lexicon (Fig. 6) was then constructed by analyzing the Borg Scale results collected from the subjective rankings and selecting those corresponding to the least required effort. For detailed description of the process for gesture lexicon construction refer to [45].

F. Hand Trajectory Classification
For each frame in the video sequence, the centroids of the face and hands were obtained from the tracking stage. The motion model for each gesture trajectory was created based on the data collected from gestures performed by ten subjects. Two of the pool of ten subjects were quadriplegic due to a cervical spinal cord injury.

Even though the trajectories for each gesture performed by different subjects or the same subject in different instances may look similar, the precise duration of each subtrajectory within the trajectory were different. To normalize the trajectories (temporal alignment), DTW was employed [46]. The velocities' components in horizontal, vertical and depth directions of both hands were selected as the feature components for each motion model [41]. The procedure to construct the motion models is described in our previous work [12].

The CONDENSATION algorithm [31] was employed to classify hand gesture trajectories in the lexicon (as in Fig. 6). It employs a set of weighted samples to fit the observed data. The original algorithm in [31] was extended to work for two hands. The original expression \( S_t = (\mu, \phi, \alpha, \rho) \) (the state at time \( t \)) was extended to \( S_t = (\mu, \phi_i, \alpha_i, \rho_i) \) where, \( \mu \) is the index of the motion models, \( \phi \) is the current phase in the model, \( \alpha \) is an amplitude scaling factor, \( \rho \) is a time dimension scaling factor, and \( i \in \{ \text{right hand, left hand} \} \). The gestures in the lexicon (as in Fig. 6) were spotted using a rest position gesture [when the subjects put their hands on the arm rest (neutral position) with no hand movement]. A dynamic motion model was created for the rest position gesture. The segment between two recognized discontinuous rest position gestures is treated as a spotted gesture.

G. Gesture Customization (One Shot Learning)
One of the objectives of our prototype follows the “came as you are” paradigm [13], where new gestures can be learned by the system automatically or by observing only one instance of it. The reason for this is to reduce the level of effort involved in the training phase of the system. In this section, we validated our approach in the context of one shot learning to assess the ability of the system to generalize learning from very few observations. A Savitzky–Golay smoothing filter [47] was added to smooth the 3-D trajectories during the creation of the
TABLE III
Hand Tracking Performance

<table>
<thead>
<tr>
<th>Method</th>
<th>False Merging (6080 Frames)</th>
<th>False Labeling (157 Interactions)</th>
<th>Tracking Accuracy (%)</th>
<th>Particle Number</th>
<th>Number of Body Parts</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCMC [41]</td>
<td>568</td>
<td>33</td>
<td>74.87</td>
<td>100</td>
<td>--</td>
</tr>
<tr>
<td>MI [23]</td>
<td>323</td>
<td>20</td>
<td>82.58</td>
<td>100</td>
<td>--</td>
</tr>
<tr>
<td>ETH (color) [24]</td>
<td>11</td>
<td>4</td>
<td>74.80</td>
<td>--</td>
<td>6</td>
</tr>
<tr>
<td>ETH (depth) [24]</td>
<td>0</td>
<td>3</td>
<td>58.25</td>
<td>--</td>
<td>6</td>
</tr>
<tr>
<td>Body Part (color) [25]</td>
<td>72</td>
<td>33</td>
<td>64.80</td>
<td>--</td>
<td>26</td>
</tr>
<tr>
<td>Body Part (depth) [25]</td>
<td>220</td>
<td>14</td>
<td>27.58</td>
<td>--</td>
<td>26</td>
</tr>
<tr>
<td>Kinect Skeleton [48]</td>
<td>1</td>
<td>4</td>
<td>73.87</td>
<td>--</td>
<td>16</td>
</tr>
<tr>
<td>CPMC (Proposed)</td>
<td>1</td>
<td>4</td>
<td>97.52</td>
<td>100</td>
<td>--</td>
</tr>
</tbody>
</table>

The tracking performance of these algorithms was evaluated by employing three metrics: false merging, false labeling, and tracking accuracy. The false merging is defined as the situation where the tracker of one hand occupies 80% of the area of the other hand. The false labeling is defined as the situation where the trackers of both hands change positions during/after interaction or occlusion. The tracking accuracy is defined by

\[
\text{Tracking Accuracy} = \frac{\text{total number of tracked frames} \times (\text{true positives} + \text{true negatives})}{\text{total number of tracked frames}}
\]

where a true positive is defined as the situation whereas a target object is present and the tracker was able to find it. True negatives are instances where the target object is not present and the tracker also agreed that the object was absent [47]. Table III shows that the proposed algorithm (CPMC) exhibits the best performance for the interaction and occlusion conditions among the three particle-based methods. There is a marginal decrease in the algorithm speed. When there is no interaction or occlusion occurring, CPMC has the same speed as MCMC and MI approaches. Comparing to the skeleton tracking [24], [48] and body part tracking method proposed by [25], the proposed method obtained higher tracking accuracy. Since our targeting user group is individuals with upper mobility impairments, two challenges exist for hand tracking in the proposed system that could not be tackled very well by skeleton-based or other articulated pose tracking methods. One challenge is that the users with upper extremity mobility impairments need to sit most of the time on a wheelchair and perform limited space hand gestures. Their hands could be very close to the body when they perform the gestures. The tracking method based purely on depth information could easily lose track when the hands are so close to the torso [24], [25]. The second challenge is that most of the wheelchairs have armrests, which can be easily confused with human arms and hands. This can explain why the skeleton-based tracking (method 3 and 5), and the articulated pose tracking (method 4) does not work well for our dataset. The values of the performance metrics “false merging” and tracking accuracy versus the distance between left and right hands are shown in Figs. 8 and 9. From Fig. 8, we can see that the proposed approach outperforms method 1, 2, 4, and
method 3 (with color images). Additionally, the result of the proposed approach (one false merging frame) was very close to the results of method 3 (with depth images) and method 5 (no false merging occurred). In Fig. 9, the total number of false positive and false negative frames versus the hands’ interdistance was presented for each method. This figure shows that the proposed approach outperformed all the other state of the art algorithms, since it displayed the fewest number of false positive and false negative frames among all the algorithms for nearly all distances.

B. Experiment 2: Gesture Recognition Performance

The motion models were constructed using the DTW algorithm. The velocities (3-D directions) of right and left hands were used as the main feature components. The gesture lexicon in Fig. 6 was adopted, and those gestures were used to create spatio-temporal trajectories that later were classified by the gesture-based recognition system.

The system was validated by eight able-bodied subjects and two subjects with quadriplegia due to cervical spinal cord injuries aged around 24–40. The ten subjects performed all the gestures in the lexicon each ten times (8 gestures x 10 subjects x 10 repetitions). Ten sessions were used for cross validation for each gesture (k-fold with k=10). In each session, 720 observations (8 gestures x 9 subjects x 10 repetitions) were used for training and 80 gestures (8 gestures x 1 subject x 10 repetitions) were used for testing. This cross validation resulted in an average accuracy of 95.9%. A confusion matrix was computed and shown in Fig. 10 (with a temporal window size of w = 19). Confusions occurred when the subjects performed a gesture mistakenly in a single direction or not enough motion was exhibited as expected in other directions. Other cases of misclassification occurred when two gestures shared similar subtrajectories (i.e., counter clock and S gestures).

The recognition performance for the CONDENSATION algorithm with our training procedures (CONDENSE) was compared to four other existing state-of-the-art recognition algorithms: 1) Basic motion [50]; 2) Motion-based PCA [51]; 3) DTW [52]; and 4) HMM [28]. After applying each gesture recognition method to our data set, the results shown in Table IV were obtained. The confusion matrices for the different methods are shown in Figs. 11, 12, 13, and 14, respectively. Method 1, 2, and 3 used motion information to recognize hand gestures, while 4 and the CONDENSATION method recognized hand gestures by extracting and classifying hand trajectories. The comparison results demonstrate a high recognition accuracy for the trajectories classification-based method. HMM-based recognition method can get comparable results as the method used in our paper.

C. Experiment 3: One Shot Learning Performance

One instance (one repetition by a subject) for each gesture in the lexicon was used for training and remaining observations were used for testing. Ten sessions were used for cross validation for each gesture (k-fold with k=10). In each session, ten observations (10 gestures x 1 subjects x 1 repetitions) were
used for training and 740 gestures (8 gestures x 9 subject x 10 repetitions and 2 gestures x 1 subject x 10 repetitions) were used for testing. This cross validation resulted in an average accuracy of 82.78%. A confusion matrix was computed and is shown in Fig. 15 (with a temporal window size of $w = 19$). The recognition accuracy found is comparable to those reported in the ChaLearn Competition [33] in 2012 (fourth place in the competition).

D. Experiment 4: Robotic Control Performance

Since the first remotely driven robotic arm developed by Goldberg et al. [53] for gardening tasks (Telegarden), there has been an extensive wave of remote labs enabling users to perform lab experiments without the need to physically attend them. Some examples include a work-cell with a 6 axes robot for conducting experiments remotely [54]; a LEGO mobile robotic platform for experimenting with autonomous navigation [55]; and a remote laboratory on robotics was developed at University of Siena called TeleTab [56].

A chemistry laboratory-based experiment was performed by five subjects including two individuals with quadriplegia due to a cervical spinal cord injury and three able-bodied individuals. In the laboratory case study experiment, a mobile robot was controlled by the gesture algorithm to transport a beaker to a position near a robotic arm. The robotic arm was activated by the operator to add a reagent to the beaker and then, the mobile robot was brought back to its original position. The gestures (a)–(h) (from the lexicon in Fig. 6) were used and mapped to the commands: change mode, robotic arm action, go forward, go backward, turn left, turn right, stop, and enable robotic arm. The two robots were controlled by three modes—discrete, continuous, and hybrid mode (discrete plus continuous mode). In discrete mode, for each issued command, the mobile robot moved a fixed increment of distance. While in continuous mode, the mobile robot responded to a given command, until the stop command was issued. To switch between the discrete and continuous mode one distinctive gesture (upward) was used. In the experiment, the discrete, continuous and hybrid (continuous plus discrete) control modes were each tested five times by all subjects. The resulting average task completion times were 241.8, 134.7, and 169.6 seconds, for the discrete,
continuous and hybrid mode, respectively (Fig. 16). From the results, the completion time of discrete mode took longer time than the continuous. Continuous and hybrid modes require commands to be issued only when the robot needs to change directions or stop, therefore fewer operations were required for continuous and hybrid modes than for discrete mode for the task observed.

V. CONCLUSION

A machine vision-based gestural interface was developed for individuals with upper extremity physical impairments. Since skin and nonskin color histogram models were used to initialize the face and hands’ centroid, the performance of the system may be affected when the users wear short sleeves. In addition, it was expected that users will be seated within the working distance to range specified by the Kinect sensor. An interaction model was incorporated into the color-based particle filter framework for hand tracking. When there was no interaction between the face and hands, multiple independent particle filters tracked the users’ movements. When interaction was present, the multiple independent particle filter trackers were combined with an interaction model to solve false merging and false labeling problems. A comparison between our proposed approach (CPMC) and five state-of-the-art algorithms demonstrated that our approach can achieve robust performance for hand tracking through interaction and occlusion conditions. The proposed tracking strategy can obtain significantly better performance than the other three methods for both false merge and false labeling problems in hand tracking through interaction and occlusion. Yet, improvements are still needed for false labeling solving. A training procedure was proposed to obtain motion models for each gesture in the lexicon. The CONDENSATION algorithm with the proposed training procedure was used and compared with four other recognition algorithms to classify bimanual gestures. Results showed that HMM-based recognition methods may deliver comparable results to our method. Thus, higher recognition could be achieved by using trajectories classification-based method. The gesture recognition algorithm designed was found to reach a recognition accuracy of 95.8%. One shot learning was applied in this paper to customize gestures and reduce the number of repetitions required to train/teach the system to a minimum (one observation). The results obtained were comparable to the state-of-the-art one shot gesture recognition algorithms presented in the ChaLearn Challenge [33].

A simulated laboratory task experiment was conducted, a typical biomedical lab procedure with the help of two robots, which were controlled through a gestural interface. Subjects with upper extremity physical impairments can successfully use the machine vision-based gestural interface to control the two robots. It was found that the proposed gestural interface was robust enough to support the completion of this task for subjects with upper extremity mobility impairments. In addition, three modes of operation were compared: discrete, continuous, and hybrid. Results showed that the continuous mode required the least average task completion time, while the discrete control mode requires the most. Therefore, the authors recommend to use continuous control mode in general, and to use discrete mode only when the robot is very near to the target, for precise location and manipulation.

VI. FUTURE WORK

Future work for this paper may include: 1) develop more effective and robust algorithms to solve false merge and false labeling problems of hand tracking through interaction and occlusion; 2) extend the laboratory task to increase the pool of participating users. Ideally, users with physical impairments...
can participate and provide feedback about the usability, learning and adaptability to the interface suggested.

APPENDIX

The video sequences of two activities for hand tracking through interaction are shown in Figs. 17 and 18.

REFERENCES

[1] M. R. Ahsan, “EMG signal classification for human computer interac-

Authorized licensed use limited to: Purdue University. Downloaded on June 25,2020 at 21:57:13 UTC from IEEE Xplore. Restrictions apply.


Hairong Jiang received the B.S. and M.S. degrees in control science and engineering from the Harbin Institute of Technology, Harbin, China, in 2008 and 2010. She is currently pursuing the Ph.D. degree at the School of Industrial Engineering, Purdue University, IN, USA.

Her current research interests include gesture recognition and assistive technology.

Bradley S. Duerstock received the B.S. degree in biomedical engineering from Purdue University, West Lafayette, IN, USA, in 1994, and the Ph.D. degree in neurobiology from the College of Veterinary Medicine, Purdue University, 1999.

He was a Post-Doctoral Research Associate with the Center for Paralysis Research, Purdue University. He is currently an Associate Professor of Engineering Practice with the Weldon School of Biomedical Engineering and School of Industrial Engineering, Purdue University. He is the Director of Institute for Accessible Science, Purdue University. His research focuses on enhancing functionality of persons with disabilities through the development of assistive technologies and accessible design and through restoration of the biomedical system.

Juan P. Wachs (M’03) received the M. Sc. and Ph.D. degrees in industrial engineering and management from the Ben-Gurion University of the Negev, Beer-Sheva, Israel.

He is currently an Assistant Professor with the School of Industrial Engineering, Purdue University, West Lafayette, IN, USA. He is the director of the Intelligent Systems and Assistive Technologies Laboratory, Toronto, ON, Canada, and is affiliated with the Regenstrief Center for Healthcare Engineering, Purdue University. He completed post-doctoral training at the Naval Postgraduate School’s MOVES Institute in the area of computer vision, under a National Research Council Fellowship from the National Academics of Sciences, and he was a recipient of the Air Force Young Investigator Award 2013. His current research interests include machine and computer vision, robotics, teleoperations, human robot interaction, and assistive technologies.

Dr. Wachs is a member of the Operation Research Society of Israel. He has published in journals including the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS, Journal of American Medical Informatics, Communications of the ACM, and the Journal of Robotic Surgery.