Hand Motion and Grasping: Capturing, Recognizing and Synthesizing

Daniel Thalmann, Hui Liang, Junsong Yuan, Singapore

ABSTRACT

In this paper we review the various previous methods in vision-based hand motion analysis and hand motion synthesis and grasping. We present our two solutions to the problem. First, we present an articulated hand pose estimation scheme, in which we improve the regression forest based methods by incorporating high-level hand part feature and hand motion constraints. Second, we present a framework for joint 6D palm pose tracking and hand gesture recognition. Based on the algorithms developed above, we have built several real-time systems to enable human-computer interaction with bare hands, such as virtual object manipulation and hand gesture recognition. We also discuss methods to generate hand motion and grasping movement for Virtual Humans.

1. INTRODUCTION

Hand motion analysis and synthesis have been a longstanding research topic in computer vision due to their importance in human-computer interaction (HCI) and film industry. Being capable of conveying rich information, hand can serve for various communicative and manipulative purposes in HCI and animation production, such as sign language recognition (Pugeault et al. 2011), virtual object manipulation (Liang et al. 2012), mechanical design (Wang et al. 2011) and realistic hand motion generation of cartoon characters (Ballan et al. 2012). A lot of solutions have been proposed in this field due to the significance of the problem. However, as the hand is highly flexible, it is generally a challenging task to capture and recognize the hand motion robustly. Up to date the most reliable methods are still the specialized hardware to record the hand motion measurement, such as the data-gloves and optical/electro-magnetic sensors. Although they can capture hand motion accurately at high frame rate, such systems are cumbersome and expensive to use.

The vision-based methods are economical alternative for specialized hardware, which capture and recognize hand motion by analyzing the visual images of hand. Fig. 1 shows a common framework for vision-based hand motion analysis. Since no instrumented devices are attached to the hand, these methods can also provide more natural and non-intrusive interaction experiences. However, the vision-based approaches have their own challenges. First, hand has high degrees of freedom, and it requires large amount of computation to seek for the optimal pose in such a high dimensional space. Also, in contrast to other articulated objects such as the human body, the hand can rotate freely in 3D space, and thus suffers from severe self-occlusion in monocular inputs, e.g. the fingers occlude each other or they are occluded by the palm. As a result, the estimated poses are often ambiguous due to the loss of the occluded parts. In addition, the environment for the HCI applications is usually uncontrolled, e.g. illumination variation, cluttered background, etc. The incurred imperfectness in hand detection and feature extraction also degrades the pose estimation accuracy. Due to these issues, the vision-based methods are inferior in accuracy compared to the specialized hardware, but the problem has been alleviated with the recent advent of commercial depth sensors and large-scale machine learning techniques, and there have been some commercialized vision-based systems, such as Leap Motion and Intel RealSense.

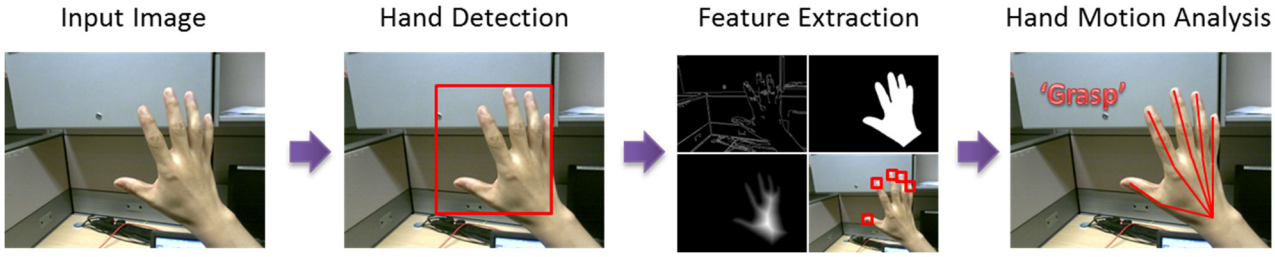

Figure 1: The pipeline for vision-based hand motion analysis.

The necessity to model interactions between an object and an autonomous virtual human (VH) appears in most applications of computer animation and Virtual Reality. Such applications encompass several domains, as for example: VHs living and working in virtual environments, VHs in emergency situations, virtual crowds, human factors analysis, training, education, virtual prototyping, and simulation-based design (Badler 1997). An example of an application using agent-object interactions is presented by Johnson et al. (1997), whose purpose is to train equipment usage in a populated virtual environment. The VH could grasp a simple object or manipulate an object, which has a role to play; a functional object or a smart object. For example, we cannot grasp a door, but we open it. Finally, before grasping an object, the VH should reach it, which can be tedious if there are many other objects or obstacles around.

Object interaction in virtual environments is also an active topic and many approaches are available in the literature. In this case, the concerned topic is the direct interaction between the user and the environment (Hand 1997; Bowman, and Hodges 1999; Poupyrev and Ichikawa 1999). The typical VR situation consists in a user trying to grasp a Virtual Object using a Cyberglove. Here, also the object could have a role to play and becomes a smart object. Moreover, to feel the object would be greatly facilitate for the user if he/she can feel it, this means to have a tactile sensation or/and a force feedback effect.

In this paper we review the various previous methods in vision-based hand motion analysis and hand motion synthesis and grasping. We present our two solutions to the problem. First, we present an articulated hand pose estimation scheme, in which we improve the regression forest based methods by incorporating high-level hand part feature and hand motion constraints. Second, we present a framework for joint 6D palm pose tracking and hand gesture recognition. Based on the algorithms developed above, we have built several real-time systems to enable human-computer interaction with bare hands, such as virtual object manipulation and hand gesture recognition.

The human hand is a complicated articulated structure with 27 bones. Not only must the movements of these joints be calculated, but also the reaching motion of the arm and the body needs to be considered. For real-time performance with many agents, fast collision-detection and inverse kinematics algorithms (Tolani et al. 2000) will be necessary in most cases. The calculation of the hand and body postures is not the only difficulty in grasping: realistic grasping also requires significant input about the semantics of the object.

Even if the geometric and physical constraints permit, sometimes an object is simply not grasped that way. For example, a door handle must not be grasped from the neck section if the goal is to turn it. A fully automatic grasping algorithm that only takes the geometry of the object into account cannot always come up with solutions that are satisfactory in this sense. Fortunately, the grasping problem for virtual characters is easier than its robotics counterpart. Simply put, we do not have to be as accurate and physical constraints are much less of a problem. The main criterion is that the grasp must look realistic.

In fact, the apparent physical realities of a virtual environment can be very different from those of the real world, with very different constraints being imposed. For example, we can imagine a virtual human holding an object that is several times his size and weight in air, while grasping it at a small

site on the edge. This does not conflict with the previous examples addressing the reality issue, as for an autonomous virtual human in a virtual setting this is more a question of what he intends to do with the object (semantics) than the actual physics of grasping.
Virtual humans are more and more used for various virtual reality applications and virtual humans should react with natural reactions as similar as real humans. For example, we can consider virtual object manipulation between real and virtual humans. It can be applied to collaborative designing, training and simulation process via 3D telepresence environment. In 3D telepresence environment, virtual human can collaborate together with real humans for supporting specific task. For example, a virtual trainer teaches real humans how to improve behaviours about object manipulations in the best way. Virtual humans also replace task of real participants who will be absent during collaboration. However, this area has received little attention because it consists of three different and complex research domains. Firstly, virtual object manipulation by a real human is important to decide usability of the application. Grasping and manipulation with hands should be very intuitive. When a real human can grasp a virtual object with her hands and if he/she can feel realistic haptic feedback on her hands, the user can manipulate object without knowledge to manipulate it. So that conventional research generally focused on how to provide natural grasping with realistic haptic feedback. Exoskeleton based interfaces are attached to provide real human forces of the hand and fingers. However, it is still difficult to generalize direct hand manipulation for end-user because conventional researches use expensive and cumbersome devices for manipulation and haptic feedback. Since the haptic interface uses wires to control devices, it also restricts the user's body movements. Computer vision based direct hand manipulation method was proposed for less cumbersome grasping. However, the working space is still limited and accuracy of grasping is lower than glove-based approach. Secondly, we need consider virtual object manipulation by virtual humans. To do this, methods of creating grasping and manipulation motions of virtual human has been studied in computer animation domain. In order to create real-time and natural looking motion, motion synthesizing techniques with rule based approach based on inverse kinematics are used. Although IK-based synthesis creates grasping motion for various objects with different sizes and shapes, grasping posture of hand requires more realistic motions and it also should be combined with other motion of virtual humans such as reaching and locomotion to support wide range of movement and manipulation tasks in real-time.

2. LITERATURE REVIEW

This section reviews the previous techniques in vision-based hand motion capture and recognition, including visual feature extraction, hand pose estimation and gesture recognition techniques.

2.1. Visual Features for Hand Motion Analysis

Visual features for hand motion analysis can be categorized into low-level and high-level features. The low-level features, e.g. edges, silhouette, skin texture and raw depth, are widely used for hand pose tracking and gesture recognition, which can be either used directly to infer the hand pose and gesture or be processed to obtain high-level hand features. Usually they don't require much prior knowledge of the hand shape and can be extracted relatively robustly and efficiently with basic image processing techniques. Edges and silhouettes are universal features for pose recovery in color images. Edge contains rich information about the external hand shape and the internal finger boundaries, and thus is useful to determine the finger configurations (Athitsos et al. 2003, Guan et al. 2006). However, edge alone is unreliable since the hand and finger edges are usually mixed up with the non-informative edges from the background clutter or the shading and texture on the hand in unconstrained environment. Silhouette describes the boundary of the hand region and gives coarse clues for the hand orientation and finger articulations. Skin color detection is one common technique to extract the hand

silhouette in color images (Li et al. 2013). In depth images, extracting hand silhouette is relatively easy, e.g. by depth threshold (Mo et al. 2006) or utilizing body part contexts (Wang et al. 2014). Compared to edges, silhouette is less discriminative since the inner shapes of the hand are ignored, e.g. edges of bending fingers. Thus it is mostly used as a supplementary cue to other features such as edge or depth for pose estimation (Lin et al. 2004, Oikonomidis et al. 2011). Skin texture is also useful in highly-controlled environment (Gorce et al. 2010), which can reduce matching ambiguity effectively.

Depth sensors are recent powerful tools for computer vision, and raw depth image has shown effective to recover hand pose via model-compatibility checking (Qian et al. 2014). Besides, the normalized local depth patches have also proven very successful when combined with spatial pooling (Tang et al. 2013, Liang et al. 2015). Their success is largely attributed to their simplicity, efficiency to calculate and effectiveness to capture the local structure of the articulated hand.

There are also various other low-level features to analyze hand motion, such as the shape context (Belongie et al. 2002), Fourier descriptor (Chen et al. 2003), orientation histogram (Freeman et al. 1995) and histogram of 3D facets (Zhang et al. 2013), which encode the high dimensional raw images into descriptors of much lower dimension. These features are effective in gesture recognition, but they are not commonly used for full-DOF hand pose estimation as a lot of detailed hand information is dropped during feature encoding.

High-level features such as fingertips and hand part labels are obtained by processing low-level features with domain knowledge and can better encode hand part semantics from the raw visual inputs, which are more efficient for hand movement analysis (Sridhar et al. 2013). For instance, in pose estimation, the fingertips can be used directly as the inputs of inverse kinematics solvers (Chua et al. 2002), or as subsidiary cues to enforce strong constraints to reduce the feasible hand pose space (Liang et al. 2013, Qian et al. 2014). Their trajectories can be used for gesture recognition (Oka et al. 2002). One popular way for fingertip detection is shape analysis of the hand contour (Oka et al. 2002), as the stretched fingers form protrusive parts on the contour. While such methods cannot handle complex configurations like bending fingers, researchers have proposed to encode more shape information for fingertip detection, such as Hough voting via multi-scale edge extraction (Do et al. 2011) and HoG descriptor (Sridhar et al. 2013). Geodesic distance map in the depth images is also effective for fingertip detection based on the observation that fingertips usually locate where the geodesic distances from the palm center are maximized (Krejov et al. 2013). However, robust detection of all fingertips is difficult due to the self-occlusion of the hand, and thus fingertips are often used as a subsidiary technique for hand motion analysis.

Hand part label is another high-level feature for hand motion analysis, which itself is a close representation of full-DOF hand pose and can be further combined with low-level features for more accurate pose prediction (Liang et al. 2015, Ionescu et al. 2014). A coarse labeling can be obtained by running pixel-level hand part detectors in the images (Keskin et al. 2011), but the results tend to be noisy. This can be improved by enforcing spatial and temporal smoothness. In Vela et al. (2012) graph cut optimization is adopted to improve parsing results of (Keskin et al. 2011). In Liang et al. (2014) a Superpixel Markov Random Field is proposed to build a superpixel-level graph to improve per-pixel classification. The superpixels segment the image so that the misclassified regions are isolated and depth discontinuity is well reserved, leading to improved parsing results compared to (Vela et al. 2012). Another popular technique is the deformable pictorial model (Felzenszwalb et al. 2005), which arranges the different parts in a deformable configuration and model the correlations between them as a tree structure for part label inference.

2.2. Vision-based Hand Pose Estimation

Vision-based hand pose estimation has been extensively studied in literature. Since bare hand is homogeneous in color, color markers or gloves are often adopted to aid analysis in RGB inputs (Chua

et al. 2002). However, markerless methods are more favorable since they are less intrusive and more convenient to use. Model-based fitting and template-matching are two main categories of methods for markerless hand pose estimation, in which the optimal pose is inferred in either a generative or discriminative manner respectively.

Model-based fitting methods: they are usually built upon a generative deformable hand model and seek for the optimal pose by iterative adjustment of pose parameters of the model and compatibility check between model features and input images. In Lin et al. (2004) the feasible hand configuration space is discretized and indexed with a KD-tree. The Nelder-Mead simplex algorithm is adopted to search for the hypothesized pose that best matches the input in terms of edge and silhouette similarities. In Oikonomidis et al. (2011) a Kinect depth camera is adopted to capture the hand image as it can better handle the background clutter and pose ambiguity in monocular color image, the particle swarm optimization algorithm is used to find the optimal pose that best fits the image projection of a 3D hand model to the input depth image and skin silhouette.

High-level hand features such as fingertips can be used to enforce extra constraints during model-fitting. In Ballan et al. (2012) a model-based framework is presented to capture subtle hand motion with multi-view inputs of eight HD cameras. The fingernails are detected in each view by Hough forest classifiers and used with image edges and optical flow to fit to an elaborate hand model. In Sridhar et al. (2013) the fingertips are detected in depth images by SVM classification with HoG descriptor, which are used for discriminative pose inference with offline synthesized database. Meanwhile, another pose candidate is obtained via generative model fitting. The optimal pose is chosen from the two candidates by selecting the one minimizing the matching error. A similar framework is presented in Qian et al. (2014), in which the protrusive fingertips are detected by morphological analysis in the depth image. The partial hand pose is recovered from the possible incomplete 3D fingertip positions and used for initialization for the subsequent model-fitting stage, which can help to speed up convergence as well as to avoid local optima.

Template-matching methods: the drawbacks of model-based methods are slow convergence in high dimensional pose space and sensitivity to initialization. In contrast, template-matching methods do not require the time-consuming computation of the hand model projection for matching. Besides, hand motion constraints are automatically included in the training data. Generally, these methods need to build a large dataset to cover the possible hand postures, and each template in the dataset contains certain features for matching and associated pose parameters. During testing, the input hand pose is recovered by looking for the templates that share the similar features. Such methods have been brought up quite long ago, and have gained high popularity in recent years with the advent of low-cost depth cameras to demonstrate promising results.

The various template-matching methods mainly differ in their feature encoding schemes and feature mapping strategies. In Guan et al. (2006) an isometric self-organizing map is used to learn a nonlinear mapping between image features and pose, which reduces the dataset redundancy by grouping templates with similar features and poses together. The hand edges are captured at only depth discontinuities with a multi-flash camera and encoded into shape context for matching. In Romero et al. (2009) locality-sensitive Hashing is utilized to retrieve multiple candidates from the database based on the HoG feature of the input image. The optimal pose is estimated by applying the temporal constraints on the retrieved candidates to resolve ambiguity. In Xu et al. (2010), the hand pose parameters are decomposed into many overlapping subsets. LSH-based nearest neighbor search is used to get the partial estimation for each subset, and the results are further integrated by a simulated annealing EM algorithm to estimate the global pose.

Among the discriminative methods, random forest and its variants have proven very effective for hand motion analysis in depth images (Xu et al. 2013, Tang et al. 2013). Its success can be attributed two aspects. First, it can be implemented to run very fast for both training and testing due to its suitability for parallel processing and robustness to noisy data. Especially, it has reported good performances for high-dimensional inputs (Caruana et al. 2008), which is suitable for the articulated

hand pose. Second, it is effectively complemented by the spatial voting technique to aggregate the pose predictions from spatially-distributed voting elements, so that the fused prediction can be less sensitive to imperfect inputs. In Xu et al. (2013), the authors propose to use the random forest to directly regress for the hand joint angles from depth images. With a pre-trained forest, each pixel casts its votes for the joint positions individually, and the votes from all the pixels are fused to a set of candidates. The optimal one is determined by a verification stage with a hand model. A similar regression forest base method is proposed in Tang et al. (2013), with the new characteristic that transfer learning is utilized to handle the discrepancy between synthesized and real-world data. In Kirac et al. (2013) the authors propose to utilize the regression forest to predict the hand pose, with an additional stage to apply bone length constraints to obtain the optimal pose. In Sun et al. (2015) multiple regression forests are organized in a cascaded manner, so that the predicted hand pose can be adjusted from coarse to fine.

Hybrid Methods: both methods have their pros and cons. On the one hand, model-based methods produce continuous pose predictions by gradually fitting the model to the visual inputs, but they are relatively slow to converge and sensitive to initialization. On the other hand, template-matching methods are fast and robust to initialization, but lack the power to differentiate among ambiguous pose predictions. Moreover, since the training data are generated by sampling discrete hand pose parameters, template-matching methods thus can only produce discrete pose predictions. To this end, both methods can be combined to supplement each other. For instance, model-based fitting can serve as a verification stage for template-matching by selecting the optimal pose through checking the compatibility between the model and the inputs (Xu et al. 2013). On the contrary, pose retrieval can also serve as an initialization stage for model-based fitting. In Wei et al. (2012) the random forest classifier provides rough part parsing for fitting the size of the 3D model to the real inputs as well as for initialization and recovering from tracking failure. Both methods can also be used for pose estimation independently and their predictions are finally fused up to certain criteria. In Baak et al. (2011) the geodesic extrema are extracted from the depth images, which are used to retrieve the candidate pose by searching in the database of geodesic extrema templates. Another candidate pose is obtained by fitting a mesh body model to the depth image, and the final prediction is taken to fit to both estimations.

2.3. Vision-based Hand Gesture Recognition

Hand gesture recognition can be categorized into static and dynamic gesture recognition. The static gestures, e.g. most of the American Sign Language alphabet (Pugeault et al. 2011) or the digit number gestures (Ren et al. 2013), mainly focus on analyzing the gestural information of the hand shape extracted from the visual inputs. Since the problem of detecting hand in color images for arbitrary pose itself is quite challenging, some gesture recognition methods try to circumvent hand detection by directly detecting the different gestures and treating each of them as an individual object class (Chen et al. 2007). However, such methods are confined to only a small number of gestures, and many hand detection methods have also been proposed to address this issue (Mei et al. 2015, Spruyt et al. 2012). Given the hand region is robustly detected, static gesture recognition can be regarded as standard classification tasks with emphasis on different hand shape representations and classification algorithms. In Freeman et al. (1995) an orientation histogram is proposed to record the statistics of the local orientation distribution of different hand shapes, which is invariant to transition and rotation variations. Different gestures are recognized by searching for the nearest neighbor in pre-stored templates. In Pugeault et al. (2011) the hand shape is encoded as the image responses from multi-scale Gabor filters and the random decision forest is used for gesture classification. In Ren et al. (2013) the hand contour is obtained by thresholding input depth image and the gestures are recognized by contour matching with pre-defined templates via Finger-EarthMover's Distance. In Zhang et al.

(2013) the local normal orientations of the 3D hand surface are grouped via a concentric spatial pooling scheme to describe the hand shape, and the SVM classifier is used for gesture classification. Dynamic gestures, e.g. waving hand in different directions (Kim et al. 2007), take the hand shapes and motion dynamics into account during gesture classification. Sometimes the hand shape information is totally discarded and only the hand motion trajectories are used for gesture recognition (Black et al. 1998), which can be efficiently fulfilled with the Dynamic Time Warping algorithm. However, the hand shape also conveys important semantic meaning in certain movement, and thus the hand motion trajectory only is inadequate for more complex tasks. Therefore, many dynamic hand gesture recognition systems are also built upon static gesture recognition algorithms, in which the static gestures are first recognized for each individual frames separately based on the hand shape. These isolated gesture sequence can be recognized by utilizing the temporal contexts, e.g. the finite state machine (Jo et al. 1998) and the hidden Markov model (Chen et al. 2003).

2.4. Synthesis of hand motion for grasping

Synthesis of hand motion for grasping a virtual object is a challenging area in computer animation. Cutkosky (1989) introduced first a method for robotics. Mas et al. (1997) proposed rule-based automatic hand synthesis approach to decide between one handed grasping and two handed grasping, according to size of a virtual object. Although this approach is available to grasp all or part of a virtual object, there is no guarantee that synthesized grasps motions are natural and consistent in comparison to a real grasping situation. A feasible alternative approach is to use prerecorded grasps data for grasp synthesis. Elkoura et al. (2003) utilized a database of human grasps to process kinematically synthesized hand poses with persevered natural coupling between joints. They applied this to playing musical instruments. Li et al. (2008) explored a data-driven approach to grasp synthesis by searching closet exampels in a prerecorded grasp database to match the object shape [20]. Amor et al. (2008) proposed a probabilistic model to constrain the solution space of human grasp synthesis from prerecorded data. Kyota et al. (2012) combined prerecorded grasp poses and grasp taxonomy for interactive grasp synthesis. Zhao et al. (2012) proposed prerecorded grasp poses with physics-motion control to model interactive human grasping synthesis. The physical model considered a wide variety of objects of different shapes, sizes, masses, frictions, and external perturbations. Finally the synthesis model linked with real-time interaction module with Kinect. Ciocarlie et al. (2007) proposed a grasp simulation model for computer animation. The model optimized the hand – object contact by searching in the principal component space instead of the complete finger DOF space, therefore increasing efficiency and realism. Endo et al. (2009) have described aspects of the development and application of a detailed hand model in the Dhaiba human mode.

We also need to consider synthesis of other motions besides hand to create realistic motion of grasping and manipulation of virtual objects. Kallmann et al. (2003) proposed motion planning method to synthesize of collision free motions for both arms, with automatic column control and leg flexion. They used a probabilistic inverse kinematics solver for matching pre-designed grasps. Recently, Huang and Kallman (2010) applied motor controllers with biomechanical rules to coordinate arm, spine, and leg movements to generate a full-body reaching motion including stepping. They also proposed example-based motion synthesis method which combines pre-recorded motions and IK-based upper body planner. LV et al. (2011) proposed utilizing various reaching strategies based on biomechanics to optimize and reduce dimension. The method showed that a virtual human can grasp distinct objects using computation of whole planned motion, to achieve realtime performance. Feng et al. (2012) showed optimized synthesis approach to reduce computation costs using separation of the motion blending from path planning. They also integrated synthesis of approximated grasp into the other motions. Thus, a virtual human can grasp and manipulate virtual object with synthesis of hand, arm, reaching and gazing in real-time.

3. FULL-DOF HAND POSE ESTIMATION

The purpose of full-DOF hand pose estimation is to recover the 3D positions of a set of hand joints, *i.e.* $\boldsymbol{\phi} = \{\phi_k\}$. In our previous work we have tested and developed various techniques for full-DOF hand pose estimation, and find out that the random regression forest based methods can achieve reasonably high accuracy at low computational complexity. However, regression forest usually suffers from ambiguous predictions. Therefore, we develop a system to estimate the articulated hand pose with an emphasis on reducing prediction ambiguity by exploiting the hand part correlations (Liang et al. 2015). The framework is illustrated in Fig. 2. On the one hand, we augment the raw hand image with high-level hand part feature for improved pose regression performance, which is obtained by running a trained RDF on the input depth image to parse twelve non-overlapping hand parts. On the other hand, we incorporate the hand motion constraints to refine the results from the regression forest via a multi-modal prediction fusion algorithm. These two strategies prove to improve the prediction performance considerably.

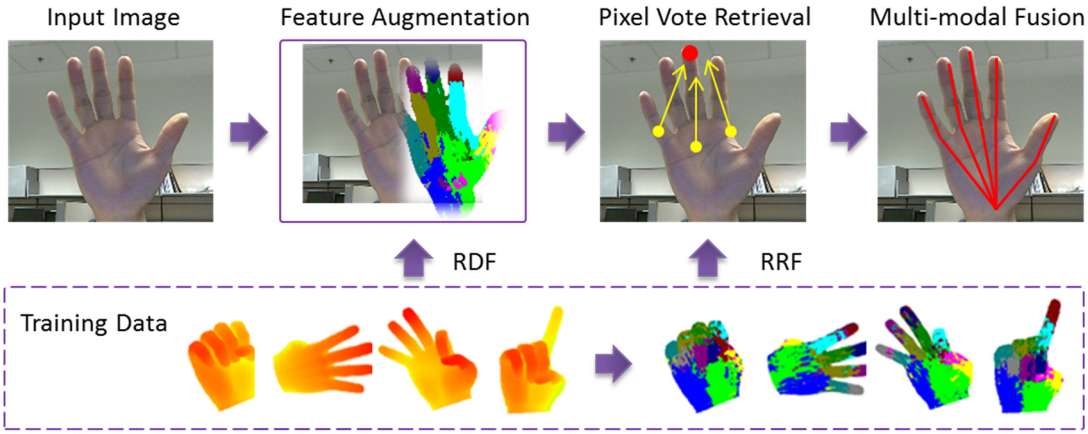

Figure 2: Full-DOF hand pose estimation framework.

Particularly, the proposed multi-modal prediction fusion algorithm is built upon the random regression forest. The random forest is an ensemble of T random decision trees, each of which is trained independently with a bootstrap training set. It is used for per-pixel hand pose retrieval in this method. During testing, it can determines a set of votes $\{\boldsymbol{v}_{ijk}, w_{ijk}\}$ for each ϕ_k for the pixel i, where $\boldsymbol{v}_{ijk}$ is the vote for the objective and w_{ijk} is the weight of the vote. When the per-pixel votes are aggregated for prediction fusion, the distribution of the per-pixel votes is generally multi-modal, and it is hard to determine which mode corresponds to the real joint position by mode-seeking for each joint separately. As the hand motion is constrained, the 3D positions of the multiple joints are highly correlated. Thus, the infeasible combinations of different joint predictions can be easily eliminated with the learned hand pose constraints. Following this idea, the ambiguous hand pose prediction can be largely resolved by Maximum a posteriori pose estimation subject to the hand motion constraints. To be specific, we perform PCA analysis to all the joint locations in the training dataset to learn a low dimensional representation of the hand configuration. During maximization of the posterior, $\boldsymbol{\phi}$ is constrained to take the linear form $\boldsymbol{\phi} = \sum_m^M \alpha_m \boldsymbol{e}_m + \boldsymbol{\mu}$, $M \ll 3 \times K$, where $\{\boldsymbol{e}_m\}$ is the set of the principal components. Besides, the joint posterior distribution $P(\boldsymbol{\phi}|I_D)$ of the entire joint set given the depth image observation can be formulated as the weighted product of the individual predictions from all the voting pixels based on the weighted Products of Experts model. In our implementation, the voting pixels are selected by randomly sampling over the entire hand region in the depth images. The task to find the optimal hand pose is thus formulated as an optimization problem:

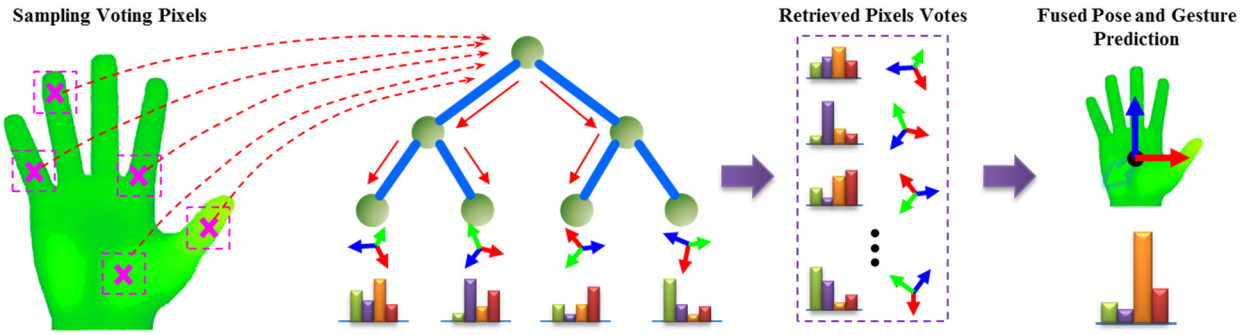

Figure 3: Joint palm pose tracking and gesture recognition with the random forest.

$$\begin{aligned} \boldsymbol{\phi}^* &= \underset{\boldsymbol{\phi}}{\operatorname{argmax}}\, P(\boldsymbol{\phi}|I_D) \\ \text{s.t. } \boldsymbol{\phi} &= \textstyle\sum_m^M \alpha_m \boldsymbol{e}_m + \boldsymbol{\mu} \end{aligned} \tag{1}$$

As shown in our work (Liang et al. 2015), this problem can be efficiently solved via an EM-like algorithm.

4. PALM POSE TRACKING & GESTURE RECOGNITION

6-DOF palm motion includes 3D translation and rotation, which is very useful to provide fully 3D immersive interaction on wearable devices, e.g. direct manipulation of virtual content in 3D. This section presents the pipeline for joint palm pose tracking and gesture recognition in depth images. For palm pose estimation, the unconstrained 6D palm motion Φ includes both rotation and translation, which are defined as the Euler angles of pitch, yaw and roll hand rotations and the 3D position of the palm center. We follow the spatial-voting based pose estimation framework and adopt the random regression forest for pose regression for the voting elements, as illustrated in Fig. 3. In this algorithm, it is used to map local pixel features to both pose and gesture votes during testing, which are then used for final fusion via spatial-voting.

In the random forest, each intermediate node has two children nodes and we store a single vote for palm pose and gesture at each leaf node. In the training stage, the random forest is trained for both the pose regression and gesture classification objectives. Given a query image I, a set of voting pixels are first uniformly sampled in the hand region and then cast their pose and gesture votes independently. Each voting pixel branches down each tree in the forest by checking the corresponding feature value until a leaf node is reached, and thus retrieves in total T votes from the leaf nodes. To obtain the final prediction, we need to aggregate the individual votes from all the voting pixels. For gesture recognition we define the gesture posterior to be the average of all the per-pixel gesture votes, and the optimal gesture is obtained by MAP estimation. For the palm pose parameters we use the Parzen density estimator to aggregate the per-pixel votes, and the optimal palm pose can be obtained efficiently by maximizing the aggregated posterior for each dimension of Φ independently with the Mean-shift algorithm. However, it is worth noting that such a basic framework does not handle ambiguous palm pose prediction quite well. In Liang et al. we have developed a novel algorithm to improve the discriminative power of the regression forest by optimizing the tree leaves, which largely improve the pose prediction accuracy of the traditional regression forest and does not require any extra computational cost.

5. HAND MOTION AND GRASPING

Grasping is perhaps the most important and complicated motion that manipulation of objects involves. The difficulty comes not only from properly "wrapping" the fingers around the object but also from the fact that the grasp must be suitable for the intended manipulation. In (Cutkosky 1989), a classification of the hand postures commonly used for grasping in manufacturing tasks is given. One of the earlier works on grasping that uses this classification to automatically generate grasping motions is the approach we described in (Mas et al. 1997). The classical taxonomy-based approach is based on three steps: first, the system uses an heuristic grasping decision based on a grasp taxonomy (Figure 4). Then, inverse kinematics is used to find the final arm posture. Finally, sphere virtual multisensors detect the contact between the hand and the object.

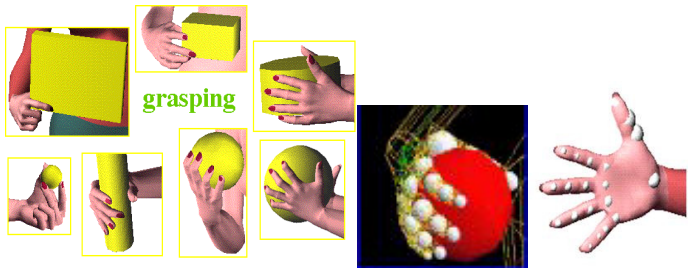

Figure 4. (a) Grasping configurations for heuristic decision (b) sensors on a ball (c) hand with sensors.

Ciger et al. (2005) introduced a new grasping framework, which brings together a tubular feature-classification algorithm, a hand grasp posture generation algorithm, and an animation framework for human-object interactions. This unique combination is capable of handling grasping tasks within the proper context of virtual human object manipulation. This is very important because how an object is to be grasped depends strongly on how it is be used. The method has the advantage that it can work with relatively complex objects, where manual approximation with simple geometrical primitives may not be possible or practical. The algorithm to detect tubular features is the Plumber algoruthm, and it is a specialized shape-classification method for triangle meshes. The Plumber method analyses the shape of an object by studying how the intersection of spheres centered at the mesh vertices evolve while the sphere radius changes. For example, for a thin limb, the curve of intersection between the mesh and a sphere will be simply connected for a small radius and then will rapidly split into two components when the radius increases and becomes greater than the tube size. While a detailed description of the shape analysis technique which uses intersecting sphere and of the Plumber method can be found in (Mortara et al. 2004), we will summarize here the main properties of Plumber and describe how the geometric parameters are associated to elongated features. First of all, Plumber can identify tubular features whose section and axis can be arbitrarily shaped, and the size of the tube is kept as a constraint during the identification process. Moreover, since the shape is analysed using a set of spheres of increasing radius, the recognition follows a multi-resolution schema.

Chosen a sphere of radius R, Plumber performs the following steps:

1. identify seed-tube regions; these regions will produce one intersection area with the sphere, with two boundary curves of intersection (see Figure 5a);
2. shrink each of the two selected intersection curves along the surface to the medial-loop, whose points are nearly equidistant from the two border loops (see Figure 5b);
3. expand-back the medial-loop by sweeping the extent of the shape in both directions. More precisely, at each iteration we place a sphere of radius R in the barycentre of the new medial loops. If the intersection between the sphere and the surface generates two loops, mesh vertices inside the sphere are marked as visited;
4. the procedure is iterated in both directions until:
   a. no more loops are found, or more than one loop is found on not-visited regions;
   b. the new loop lies on triangles that are already part of another tube, or the length of the new loop exceeds a predefined threshold.
5. the tube skeleton is extracted by joining the loops' barycentres.

(a) (b) (c) (d)

Figure 5. Plumber method: (a) identification of limb vertices, (b) extraction of their connected components and medial loop, (c) iteration, (d) tube and a cap (black) found at this scale.

After the location of seed tubular regions and the computation of the medial loop, the tubes are recovered by expanding the loop by controlled procedure which, at each step, extends the center-line and at the same time ensures that the surface is tubular around it. Finally, the barycentres of the medial loops are joined to define the tube skeleton. Our real-time grasping algorithm is based on approximating the parts of a tubular section and the finger segments with capsules (Figure 6). A capsule (or capped cylinder) is the set of points at a fixed distance from a line segment. Two capsules intersect if and only if the distance between capsule line segments is smaller or equal to the sum of the capsule radii. Given a finger segment and a tubular region, we first find out which part of the tubular region is most likely to intersect with the finger segment. We accomplish this by intersecting the finger plane with each tube center line segment. Figure 7 shows examples.

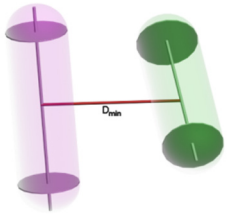

Figure 6. Capsule intersection test.

Figure 7. Examples of tubular grasping.

6. ACKNOWLEDGMENT

Part of this research, which is carried out at BeingThere Centre, is supported by the Singapore National Research Foundation under its International Research Centre @ Singapore Funding Initiative and administered by the IDM Programme Office.

7. REFERENCES

Amor H. B., Heumer G., Jung B. and Vitzthum A., Grasp synthesis from low-dimensional probabilistic grasp models. Computer Animation and Virtual Worlds, 19 (3-4), pp. 445-454, 2008.

Athitsos V. and Sclaroff S., Estimating 3d hand pose from a cluttered image. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp. 432-442, 2003.

Baak A., Muller M., Bharaj G., Seidel H. P. and Theobalt C., A Data-Driven Approach for Real-Time Full Body Pose Reconstruction from a Depth Camera. In: Proc. IEEE Int'l Conf. on Computer Vision, pp. 1092-1099, 2011.

Badler N. I., "Virtual Humans for Animation, Ergonomics, and Simulation". IEEE Workshop on Non-Rigid and Articulated Motion, Puerto Rico, June 1997.

Ballan L., Taneja A., Gall J., van Gool L. and Pollefeys M., Motion Capture of Hands in Action using Discriminative Salient Points. In: Proc. European Conf. on Computer Vision, pp. 640-653, 2012.

Belongie S., Malik J. and Puzicha J., Shape matching and object recognition using shape contexts. IEEE Trans. Pattern Analysis and Machine Intelligence, 24 (4), pp. 509-522, April 2002.

Black M. and Jepson A., Recognition Temporal Trajectories using the Condensation Algorithm. In: Proc. IEEE Int'l Conf. on Automatic Face and Gesture Recognition, pp. 16-21, 1998.

Bowman D. and Hodges L., Formalizing the Design, Evaluation, and Application of Interaction Techniques. Journal of Visual Languages and Computing, 10, pp. 37-53, 1999.

Caruana R., Karampatziakis N. and Yessenalina A., An empirical Evaluation of Supervised Learning in High Dimensions. In: Proc. Int'l Conf. on Machine Learning, pp. 96-103, 2008.

Chen F. S., Fu C. M. and Huang C. L., Hand gesture recognition using a real-time tracking method and hidden Markov models. Image Vision Computing, 21 (8), pp. 745-758, Aug. 2003.

Chen Q., Georganas N. D. and Petriu E. M., Real-time Vision-based Hand Gesture Recognition Using Haar-like Features. In: Instrumentation and Measurement Technology Conference, pp. 1-6, 2007.

Chua C.-S., Guan H. and Ho Y. K., Model-based 3D hand posture estimation from a single 2D image. Image and Vision Computing, 20 (3), pp. 191-202, 2002.

Ciger J., Abaci T. and Thalmann D., Planning with Smart Objects. In: Proc. WSCG'2005

Ciocarlie, M., Goldfeder, C. and Allen P., Dimensionality reduction for hand – independent dexterous robotic grasping. In: IEEE – RSJ International Conference on Intelligent Robots and Systems, 2007, pp. 3270-3275.

Cutkosky M. R., On grasp choice, grasp models, and the design of hands for manufacturing tasks. IEEE Transactions on Robotics and Automation, 5 (3), pp. 269-279, 1989.

de La Gorce M., Fleet D. J. and Paragios N., Model-Based 3D Hand Pose Estimation from Monocular Video. IEEE Trans. Pattern Analysis and Machine Intelligence, 33 (9), pp. 1793-1805, Sept. 2011.

Do M., Asfour T. and Dillmann R., Particle filter-based fingertip tracking with circular Hough transform features. In: Proc. IAPR Conf. on Machine Vision Applications, pp. 471-474, 2011.

ElKoura G. and Singh K., Handrix: animating the human hand. In: Proceedings of the 2003 ACMSIGGRAPH/Eurographics symposium on Computer animation, 2003, pp.110-119.

Endo, Y., Kanai, S., Miyata, N., Kouchi, M., Mochimaru, M., Konno, J., Ogasawara, M. and Shimokawa, M., Optimization – Based Grasp Posture Generation Method of Digital Hand for Virtual Ergonomics Assessment. SAE International Journal of Passenger Cars – Electronic and Electrical Systems, 1 (1), pp. 590-598, 2009.

Felzenszwalb P. F. and Huttenlocher D. P., Pictorial Structures for Object Recognition. International Journal of Computer Vision, 61 (1), pp. 55-79, Jan. 2005.

Feng A. W., YXu Y. and Shapiro A., An example-based motion synthesis technique for locomotion and object manipulation. In: Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games, 2012, pp. 95-102. ACM, 2012.

Freeman W. T. and Roth M., Orientation Histograms for Hand Gesture Recognition. In: Int'l. Workshop on Automatic Face and Gesture Recognition, pp. 296-301, 1995.

Guan H., Feris R. S. and Turk M., The Isometric Self-Organizing Map for 3D Hand Pose Estimation. In: Proc. IEEE Int'l Conf. on Automatic Face and Gesture Recognition, 263-268, 2006.

Hand C., A Survey of 3D Interaction Techniques. Computer Graphics Forum, 16 (5), 269-281, 1997.

Hernandez-Vela A., Zlateva N., Marinov A., Reyes M., Radeva P., Dimov D. and Escalera S., Graph Cuts Optimization for Multi-Limb Human Segmentation in Depth Maps. Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp. 726-732, 2012.

Huang Y. and Kallmann M., Motion parameterization with inverse blending. Motion in Games, pp. 242-253.

Ionescu C., Carreira J. and Sminchisescu C., Iterated Second-Order Label Sensitive Pooling for 3D Human Pose Estimation. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp. 1661-1668, 2014.

Jo K., Kuno Y., Shirai Y., Manipulative Hand Gestures Recognition Using Task Knowledge for Human Computer Interaction. In: Proc. IEEE Int'l Conf. on Automatic Face and Gesture Recognition, pp. 468-473, 1998.

Johnson W. L. and Rickel J., "Steve: An Animated Pedagogical Agent for Procedural Training in Virtual Environments". Sigart Bulletin, ACM Press, 8 (1-4), pp. 16-21, 1997.

Kallmann M., Aubel A., Abaci T. and Thalmann D., Planning collision-free reaching motions for interactive object manipulation and grasping. Computer Graphics Forum, Vol. 22, Wiley, 2003, pp. 313-322.

Keskin C., Kirac F., Kara Y. E. and Akarun L., Real-time hand pose estimation using depth sensors. In: Proc. IEEE Int'l Conf. on Computer Vision Workshops, pp. 1228-1234, 2011.

Kim T. K., Wong S. F., Cipolla R., Tensor Canonical Correlation Analysis for Action Classification. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp. 1-8, 2007.

Kirac F., Kara Y. E. and Akarun L., Hierarchically constrained 3D hand pose estimation using regression forests from single frame depth data. Pattern Recognition Letters, 50 (1), pp. 91-100, 2013.

Krejov P. and Bowden R., Multi-touchless: Real-Time Fingertip Detection and Tracking Using Geodesic Maxima. In: Proc. IEEE Int'l Conf. on Automatic Face and Gesture Recognition, pp. 1-7, 2013.

Kyota F. and Saito S., Fast grasp synthesis for various shaped objects. Computer Graphics Forum, 31, Wiley, 2012, pp.765-774.

Li C. and Kitani K. M., Pixel-level Hand Detection in Ego-Centric Videos. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp. 3570-3577, 2013.

Liang H., Yuan J. and Thalmann D., 3D Fingertip and Palm Tracking in Depth Image Sequences. In: Proc. ACM Int'l Conf. on Multimedia, pp. 785-788, 2012.

Liang H., Yuan J. and Thalmann D., Parsing the Hand in Depth Images. IEEE Trans. Multimedia, 16 (5), pp. 1241-1253, Aug. 2014.

Liang H., Yuan J. and Thalmann D., Resolving Ambiguous Hand Pose Predictions by Exploiting Part Correlations. In: IEEE Trans. Circuits and Systems for Video Technology, 2015.

Liang H., Yuan J. and Thalmann D., Spatially-Optimized Hough Forest for Robust Palm Pose Tracking with Arbitrary Hand Postures. In: IEEE Trans. Multimedia, under review.

Liang H., Yuan J., Thalmann D. and Zhang Z., Model-based Hand Pose Estimation via Spatial-temporal Hand Parsing and 3D Fingertip Localization. Visual Computer Journal, 29 (6-8), pp. 837-848, June 2013.

Lin J. Y., Wu Y. and Huang T. S., 3D Model-Based Hand Tracking Using Stochastic Direct Search Method. In: Proc. IEEE Int'l Conf. on Automatic Face and Gesture Recognition, pp. 693-698, 2004.

Lv P., Zhang M., Xu M., Li H., Zhu P. and Pan Z., Biomechanics-based reaching optimization. The Visual Computer Journal, 27 (6-8), 2011, pp. 613-621.

Magnenat-Thalmann N., Laperriere R. and Thalmann D., Joint-dependent local deformations for hand animation and object grasping. In: Proceedings of Graphics Interface ’88, pp. 26-33, June 1988.

Mas R., Boulic R., Thalmann D., Extended grasping behavior for autonomous human agents. In: AGENTS ’97: Proc. 1st international conference on Autonomous agents, ACM, NY, 1997, pp. 494-495.

Mei K., Zhang J., Li G., Xi B., Zheng N., Fan J., Training more discriminative multi-class classifiers for hand detection. Pattern Recognition, 48 (3), pp. 785-797, 2015.

Li M., Gao S., Fuh J. Y. H. and Zhang Y. F., Replicated concurrency control for collaborative feature modelling: A fine granular approach. Computers in Industry, 59 (9), pp. 873-881, 2008.

Mo Z. and Neumann U., Real-time Hand Pose Recognition Using Low-Resolution Depth Images. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp. 1499-1505, 2006.

Mortara M., Patane G., Spagnuolo M., Falcidieno B. and Rossignac J., Plumber: a method for a multi-scale decomposition of 3d shapes into tubular primitives and bodies. In: Ninth ACM Symposium on Solid Modeling and Applications SM’04, pp. 339-344, 2004.

Oikonomidis I., Kyriazis N. and Argyros A. A., Efficient model-based 3D tracking of hand articulations using Kinect. In: Proc. British Machine Vision Conference, 101.1-101.11, 2011.

Oka K., Sato Y. and Koike H., Real-Time Fingertip Tracking and Gesture Recognition. IEEE Computer Graphics and Applications, 22 (6), pp. 64-71, 2002.

Poupyrev I. and Ichikawa T., “Manipulating Objects in Virtual Worlds: Categorization and Empirical Evaluation of Interaction Techniques”. Journal of Visual Languages and Computing, 10, pp. 19-35, 1999.

Pugeault N. and Bowden R., Spelling It Out: Real-Time ASL Fingerspelling Recognition. In: IEEE Workshop on Consumer Depth Cameras for Computer Vision, pp. 1114-1119, 2011.

Qian C., Sun X., Wei Y., Tang T. and Sun J., Realtime and Robust Hand Tracking from Depth. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp. 1106-1113, 2014.

Ren Z., Yuan J., Meng J. and Zhang Z., Robust Part-based Hand Gesture Recognition using Kinect Sensor. In: IEEE Trans. Multimedia, 15 (5), pp. 1110-1120, 2013.

Romero J., Kjellstrom H. and Kragic D., Monocular Real-Time 3D Articulated Hand Pose Estimation. In: Proc. Int'l Conf. on Humanoid Robots, pp. 87-92, 2009.

Spruyt V., Ledda A. and Philips W., Real-time hand tracking by invariant hough forest detection. In: Proc. IEEE Int'l Conf. on Image Processing, pp. 149-152, 2012.

Sun X., Wei Y, Liang S., Tang X. and Sun J., Cascaded Hand Pose Regression. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, 2015.

Tang D., Yu T. H. and Kim T.-K., Real-time Articulated Hand Pose Estimation using Semi-supervised Transductive Regression Forests. In: Proc. IEEE Int'l Conf. Computer Vision, pp. 3224-3231, 2013.

Tolani D., Goswami A. and Badler N., Real-time inverse kinematics techniques for anthropomorphic limbs. Graph. Models Image Process 62 (5), 353-388, 2000.

Wang C., Liu Z. and Chan S. C., Superpixel-Based Hand Gesture Recognition with Kinect Depth Camera. IEEE Trans. Multimedia, 17 (1), pp. 29-39, 2014.

Wang R. Y., Paris S. and Popovic J., 6D Hands: Markerless Hand Tracking for Computer Aided Design. In: Proc. ACM Symposium on User Interface Software and Technology, pp. 549-558, 2011.

Wei X., Zhang P. and Chai J., Accurate Real-time Full-body Motion Capture Using a Single Depth Camera. ACM Transactions on Graphics, 31 (6), 188, 2012.

Xu C. and Cheng L., Efficient Hand Pose Estimation from a Single Depth Image. In: Proc. IEEE Int'l Conf. on Computer Vision, pp. 3456-3462, 2013.

Xu J., Wu Y. and Katsaggelos A., Part-based Initialization for Hand Tracking. In: Proc. IEEE Int'l Conf. on Image Processing, pp. 3257-3260, 2010.

Zhang C., Yang X. and Tian Y., Histogram of 3D Facets: A Characteristic Descriptor for Hand Gesture Recognition. In: Proc. IEEE Int'l Conf. on Automatic Face and Gesture Recognition, pp. 1-8, 2013.

Zhao W., Chai J. and Xu Y.-Q., Combining marker-based mocap and rgbd camera for acquiring high-fidelity hand motion data. In: Proc. ACM SIGGRAPH/Eurographics Symposium on Computer Animation, pp. 33-42, 2012.