

A Two-Step Minimization Algorithm For Model-Based Hand Tracking

O.Ben Henia
Liris-CNRS UMR 5205
Université de Lyon1
France
obenheni@liris.cnrs.fr

M. Hariti
Liris-CNRS UMR 5205
Université de Lyon1
France
mhariti@liris.cnrs.fr

S. Bouakaz
Liris-CNRS UMR 5205
Université de Lyon1
France
sbouakaz@liris.cnrs.fr

ABSTRACT

Model-based methods to the tracking of an articulated hand in a video sequence could be divided in two categories. The first one, called stochastic methods, uses stochastic filters such as kalman or particle ones. The second category, named deterministic methods, defines a dissimilarity function to measure how well the hand model is aligned with the hand images of a video sequence. This dissimilarity function is then minimized to achieve the hand tracking. Two well-known problems are related to the minimization algorithms. The first one is that of local minima. The second problem is that of computing time required to reach the solution. These problems are compounded with the large number of degrees of freedom (DOF) of the hand (around 26). The choice of the function to be minimized and that of the minimization process can be an answer to these problems. In this paper two major contributions are presented. The first one defines a new dissimilarity function, which gives better results for hand tracking than other well-known functions like the directed chamfer or hausdorff distances. The second contribution proposes a minimization process that operates in two steps. The first one provides the global parameters of the hand, i.e. position and orientation of the palm, whereas the second step gives the local parameters of the hand, i.e. finger joint angles. Operating in two stages, the proposed two-step algorithm reduces the complexity of the minimization problem. Indeed, it seems more robust to local minima than a one-step algorithm and improves the computing time needed to get the desired solution.

Keywords: Hand tracking, minimization algorithm, dissimilarity function.

1 INTRODUCTION

Hand gestures take a fundamental role in inter-human communication of daily life. Their use has become an important part of human-computer interaction in the two last decades [WH99][Tos06][Ste04]. Data gloves are commonly used as input devices to capture and track human hand motion by attaching some sensors to the hand to measure the joint angles and the spatial positions of the hand directly. Unfortunately, glove-based devices are expensive, frail and not user friendly.

Vision-based approaches offer promising alternatives to capture and track human hand motion with affordable cameras. However, building a fast and effective vision-based hand motion tracker is challenging. This is due to the high dimensionality of the pose space, the ambiguities due to occlusion, the lack of visible surface texture and the significant appearance variations due to shading.

In this paper, a parametric hand model is used. Articulated hand motion is decoupled to its global hand motion and local finger motion, in which global motion is parameterized as the rotation and translation of the palm, and local motion is parameterized as the joint angles of the hand. The hand motion tracking is first formulated as an optimization task, where a dissimilarity function between the projection of the hand model under articulated motion and the observed image features, is to be minimized. A two-step iterative algorithm is then proposed to minimize this dissimilarity function. This is achieved by means of a simplex approach proposed by Nelder and Mead [NM65].

The next section presents a brief literature survey of hand motion tracking. Section 3 describes the used 3D hand model as well as the dissimilarity function between the projection of the hand model and the corresponding image features. Section 4 details the hand motion tracking algorithm. Before concluding, experimental results from synthetic and real data are presented in section 5.

2 LITERATURE REVIEW

Approaches to the tracking of hand motion could be divided into two classes. In the first class, view-based approaches are usually used to recover hand pose through classification or regression techniques [RASS01][SKS01]. A set of hand features is labeled

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

with a particular hand pose, and a classifier is trained from this data. Due to the high dimensionality of the space spanned by possible hand poses, it is difficult, or even impossible to perform dense sampling. Thus, these approaches are well suited for rough initialization or recognition of a limited set of predefined poses. This family of approaches is often used in applications where computing time takes precedence over tracking accuracy. This is the case of a human computer interface (HCI) application where only few hand poses are considered, e.g. a real-time hand gesture recognition system using classifiers was proposed in [IKS07].

The second class incorporates model-based approaches. These approaches use an articulated three-dimensional (3D) hand model and provide more precise hand tracking. The underlying kinematic structure is usually based on the biomechanics of the hand. Therefore, the 3D hand model is often represented as a hierarchical one with approximately 26 degrees of freedom (DOF) [KX06][WLH05].

Models that have been used for tracking are based on planar patches [WLH01], deformable polygon meshes [HH96] or generalized cylinders [DF98]. Stenger et al. [SMC01a] used quadric surfaces like cones and ellipsoids. Other authors proposed a fine mesh obtained by scanning the hand to track [BKMM*04]. However, even if the fine mesh could increase the tracking accuracy, it remains restrictive to scan the hand to track.

A model-based tracking system uses a geometric hand model whose projection is registered to the observed image. An error function between image features and model projection is computed, and the parameters of the hand model are then adapted such that this cost is minimized. Tracking in such a system is performed by means of methods that could be classified into two categories, stochastic and deterministic ones. Stochastic methods exploit the state space to seek an estimation of the hand poses. Stenger et al. [SMC01a] used an Unscented Kalman Filter (UKF) to update the model projection pose to minimize the geometric error between the model projection and a video sequence on the background. They argued that their approach is well adapted for the hand tracking problem due to the nonlinear nature of the latter arising from the DOF of rotation. The particle filter, also known as the Condensation algorithm [IB98] in the computer vision community, has been applied to the hand tracking problem in [KX06]. It goes beyond the unimodal Gaussian assumption of the Kalman filter by approximating arbitrary distributions with a set of random samples. The advantage is that the particle filter can deal with clutter and ambiguous situations more effectively, by not placing its bet on just one hypothesis. However, a major concern is that the number of particles required increases exponentially with the dimension of the state space.

On the other hand, deterministic methods typically track by performing an iterative minimization of a cost function which measures how well the model is aligned with the images. Rehg and Kanade [RK94][RK95] proposed two possible cost functions, the first based on edge features [RK94], the second on template registration [RK95]. In the first method, they used the Gauss-Newton algorithm to minimize their edge-based cost function. The second method of template registration was used to deal with self-occlusion. Using two cameras and dedicated hardware, their DigitEyes tracking system achieved a rate of 10 frames per second when using local edge features.

Ouhaddi and Horain [OH99] suggested two cost functions. The first is defined as a non-overlapping surface between the model silhouette and the hand one extracted from an image, whereas the second as a distance between the model and image contours. For the two methods, they used three standard optimization approaches to minimize their cost functions, named Levenberg-Marquardt, downhill simplex [NM65] and Powell approaches. They obtained the best results for minimizing the non-overlapping surface by means of the downhill simplex approach. Using a single camera, they tracked simple hand motion like translation of the hand and abduction/adduction of fingers.

Delamarre and Faugeras [DF98] pursued a stereo approach to hand tracking. A stereo correlation algorithm was used to estimate a dense depth map. Note that deterministic methods do not escape from the problem of local minima. Bray et al. [BKMSG04] proposed an optimization method for hand tracking based on Stochastic Meta-Descent (SMD), which is a gradient descent with local step size adaptation that combines rapid convergence with excellent scalability. However, this method requires the knowledge of the derivative of the function to be minimized.

Our proposed work belongs to the class of model-based approaches for hand tracking. Indeed, we use a parametric hand model which is defined as a hierarchical one with 26 DOF. To reduce our model to 22 DOF, we impose constraints on the finger motion, such as linear dependence of finger joint angles. Furthermore, the state space of the hand is decoupled into global pose parameters and finger motion parameters. These sets of parameters are estimated in a two-step iterative algorithm. Specifically, a new dissimilarity function between the hand images and the projection of the hand model is defined and then minimized using a simplex approach proposed by Nelder and Mead [NM65].

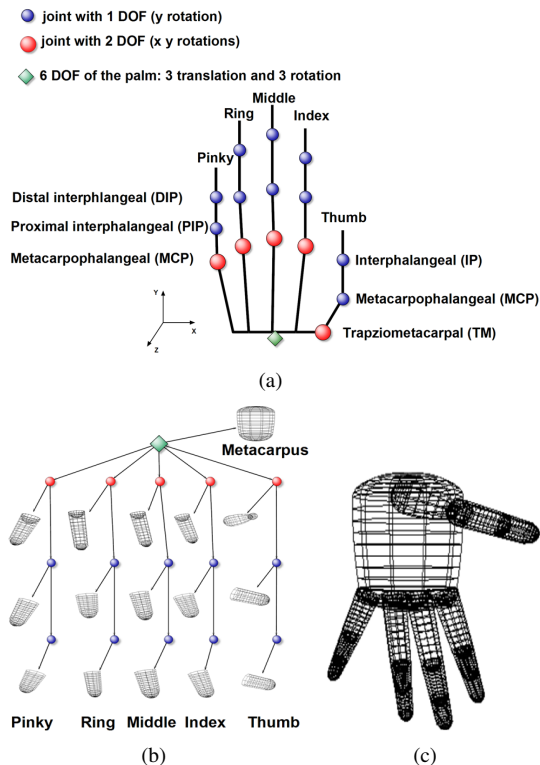


Figure 1: (a) Skeletal representation of the 3D model (b) Hierarchical representation of the 3D model (c) 3D hand model appearance

3 MODEL AND DISSIMILARITY FUNCTION

3.1 The Hand Model

The human hand is a complex and highly articulated structure. Several models have been proposed in the literature to represent the latter. In [HH96] a 3D deformable point distribution model was implemented. This model can not accurately reproduce all realistic hand motion. Indeed, since this model is not based on a rigid skeleton, fingers can be warped and reduced to ensure tracking of the hand gestures.

On the other hand, skeleton animation based model was used in [BKMSG04][OH99]. This kind of models is usually defined as hierarchical transformations representing the DOF of the hand: position and orientation of the palm, joint angles of the hand (Fig:1(a)). The variation in the values of DOF animates the 3D hand model. Using this kind of models, we can estimate not only the position and orientation of the hand, but also the joint angles of fingers.

In our proposed work, we use a parametric hand model which is conforming to the H-Anim standard¹. The H-Anim model is often used in the 3D animation

¹ Humanoid-Animation(H-ANIM) is an approved ISO standard for humanoid modeling and animation. website :www.h-anim.org

field. We can highlight its particularity to separate the kinematic part (motion) from the appearance one. This model consists of a hierarchy of 3D transformations (rotation, translation) allowing easy control of its animation by modifying only the involved transformations. Regarding the appearance, objects called segments are placed in this hierarchical representation (Fig.1(b)) to provide the shape of this model (Fig.1(c)). To change the appearance of this model, we modify the objects representing the latter. In our proposed work, these object segments are modeled by quadric surfaces as shown in Fig.1(c).

In our hand model, the base is a palm and five fingers are attached to the palm. The palm is modeled as a kinematic structure with six DOF. Three of six DOF correspond to the position of the palm. The other three DOF correspond to the orientation of the palm. Each finger is modeled as a four DOF kinematic chain. Two of four DOF correspond to the metacarpophalangeal joint (MCP) and its abduction (see Fig.1(a)). Note that for the thumb these two DOF correspond to the trapeziometacarpal joint and its abduction (see Fig.1(a)). The other two DOF correspond to the proximal interphalangeal joint (PIP) and the distal interphalangeal joint (DIP) (see Fig.1(a)).

Therefore, our hand model has 26 DOF. To reduce this number of DOF, we apply a constraint defined by Lin et al. [LWH00], which consists in exploiting dependencies between the angles of the proximal interphalangeal joint (PIP) and the distal interphalangeal joint (DIP) : $\theta_{DIP} = 2/3\theta_{PIP}$. Applied to our hand model, this constraint yields 22 DOF to be estimated.

For the model appearance, we use a set of quadrics approximately representing anatomy of a real human hand (Fig.1(c)). The palm is modeled using a truncated ellipsoid, its top and bottom closed half-ellipsoids. Each finger is composed by three truncated cones, i.e. one for each phalanx. Hemisphere was used to close each truncated cone. The major advantage of this model shape representation is its simplicity to be adapted to any hand to track compared with models based on 3D scans.

3.2 Dissimilarity Function

Many model-based approaches have been proposed in the literature to achieve low-cost monocular hand tracking. These approaches make use of a dissimilarity function, which measures how well the 3D hand model is aligned with the hand images. The most often used dissimilarity functions exploit either silhouette or edge features extracted from the images of a video sequence. The directed chamfer distance [Bor88] is a famous function, which computes a dissimilarity between two edge images I_a and I_b as a distance d_C between two edge pixel sets A and B of I_a and I_b , respectively. The dis-

tance d_C from set A to B with respect to metric d is defined as:

$$d_C(A,B) = \frac{1}{|A|} \sum_{a_i \in A} \min_{b_j \in B} d(a_i, b_j) \quad (1)$$

where $|A|$ is the cardinal number of the pixel set A and $d(a_i, b_j)$ the Euclidian distance between a_i and b_j . Another well-known dissimilarity function is the directed hausdorff distance d_H , which is proposed in [HKKR93] and defined as the maximum of all distances from each edge pixel in set A to its nearest neighbor in set B:

$$d_H(A,B) = \max_{a_i \in A} \{ \min_{b_j \in B} d(a_i, b_j) \} \quad (2)$$

where the Euclidian distance $d(a_i, b_j)$ is identical to the one described in Eq.1. A disadvantage of the edge-based dissimilarity functions is their sensitivity to outliers. Indeed, few pixels are only considered to carry out the tracking of articulated objects. On the other hand, Ouhaddi and Horain [OH99] proposed a non-overlapping function which makes use of the hand silhouette. Their dissimilarity function could be defined as the uncommon area of two surfaces: the hand silhouette and the model projection.

The above dissimilarity functions were used to track simple hand motion in a video sequence composed from images acquired by a single camera. Indeed, Ouhaddi and Horain [OH99] used their non-overlapping function to track some global hand motion such as the position of the palm, or simple local finger motion like abduction/adduction, but not local and global motion together. Stenger et al [SMC01b] evaluated their algorithm for tracking three global DOF of the hand (two translations and one rotation).

To achieve more robust monocular hand tracking, we propose a new dissimilarity function which combines the non-overlapping surface and the directed chamfer distance. Our dissimilarity function assesses the difference between the model projection and the hand silhouette. Let H_s and M_s be the respective hand silhouette (Fig.2(a)) and model projection M_p (Fig.2(b)). Let NOS be the $N_r \times N_c$ image containing the non-overlapping surface (Fig.2(d)), where N_r and N_c are the numbers of rows and columns, respectively. A pixel (i,j) of this image is defined as:

$$NOS_{ij} = \begin{cases} 1 & \text{if the pixel (i,j) belongs to} \\ & ((H_s \cup M_s) - (H_s \cap M_s)) \\ 0 & \text{otherwise} \end{cases}$$

The size of the non-overlapping surface evaluates the dissimilarity between the hand image and the model projection. To achieve more tracking robustness, we propose to weight the NOS's pixels by their distance to the hand contours. For this purpose, we extract first the

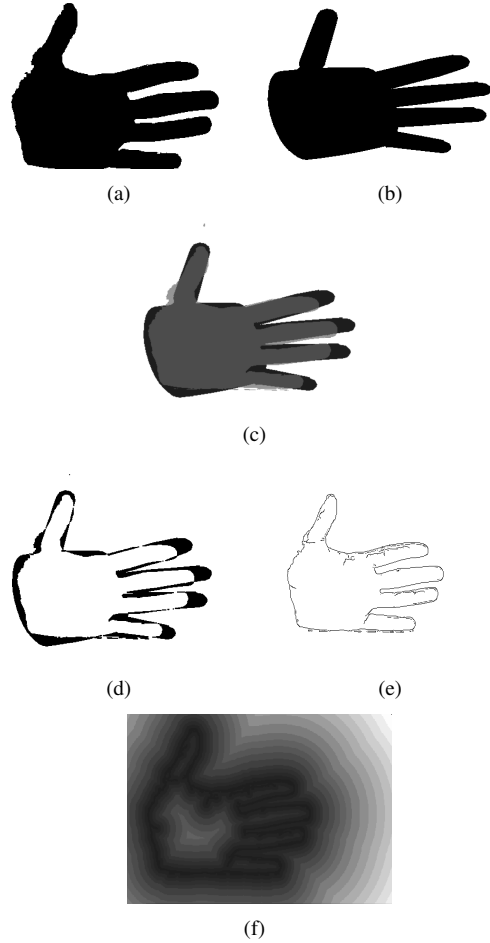


Figure 2: The different computing steps for our dissimilarity function. (a) Hand silhouette H (b) Model projection M_p (c) Superposition of the model projection and hand silhouette (d) Non-overlapping surface NOS (e) Hand contours (f) Distance map D

hand contours (Fig.2(e)) from a hand image H, and an $N_r \times N_c$ distance map D (Fig.2(f)) is then computed. An element D_{ij} of D represents the distance of the pixel (i,j) to the hand contours. We define our dissimilarity function which compares a hand image H with the model projection associated with the parameters R , T and θ , where R and T represent the global motion (three rotations and three translations) and θ represents the local motion (fingers articulation angles). Our dissimilarity function d_F is defined as follows:

$$d_F(NOS, D) = \sum_{i=1, j=1}^{N_r, N_c} NOS_{ij} * D_{ij} \quad (3)$$

4 THE TRACKING ALGORITHM

The tracking of the hand gestures in a video sequence is performed by seeking the parameters of the hand model which reproduce the hand motion. We achieve this step

by minimizing our dissimilarity function for each frame of the video sequence. This minimization gives the parameters of the hand model which align the pose model with hand pose. We assume that the parameters of the hand model are close to the solution associated with the first frame of the video sequence. For the remainder of the video sequence, the minimization algorithm exploits the parameters of the hand model obtained at the previous frame. We use the Nelder-Mead method [NM65] which is an iterative minimization algorithm. This choice is justified by two main reasons. Firstly, the use of this method does not require the knowledge of the derivative of the function to be minimized. The second reason relates to the processing of the algorithm itself. Indeed, the simplex descent explores different directions for each iteration. These various explorations can be achieved in parallel to increase computation time.

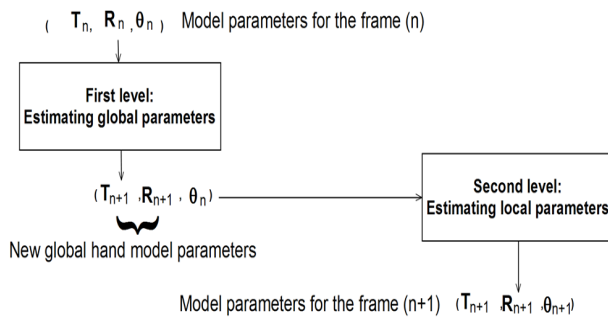


Figure 3: Two-step estimation process

Due to the high dimensionality of the search space, we decouple the minimization algorithm in two steps(Fig.3). The first one estimates the position and orientation of the hand. The parameters representing the joints angles of fingers are fixed, whereas those representing the position and orientation of the palm are processed by the minimization algorithm. The processing is reversed at the second step, i.e. the orientation and position parameters are first fixed to those obtained in the first step, and the joint angles of the hand are then estimated. Besides simplifying the minimization problem, this approach can be justified by the slow variation of the hand pose in two successive frames.

5 EXPERIMENTAL RESULTS

The performance of the two-step iterative algorithm is first evaluated for tracking hand motion appearing in synthetic and real images. We evaluate also our dissimilarity function by comparing it with the non-overlapping function. For the first test benchmark, a video sequence of one hundred 320x240 synthetic images of the hand model is acquired (Table 1). To obtain

| | Tracking algorithm | Frame 1 | Frame 50 | Frame 100 |
|--------------|--------------------|---------|----------|-----------|
| Ground Truth | | | | |
| d_C | One-Step | | | |
| | Two-Step | | | |
| d_H | One-Step | | | |
| | Two-Step | | | |
| d_{NOS} | One-Step | | | |
| | Two-Step | | | |
| d_F | One-Step | | | |
| | Two-Step | | | |

Table 1: One-Step and Two-Step tracking algorithm results using different dissimilarity functions : directed chamfer distance(d_C) directed hausdorff distance(d_H), non-overlapping function(d_{NOS})and our dissimilarity function (d_F).

this sequence of images we vary three global parameters and four local ones. The global parameters are the X-translation, Z-translation and Z-rotation of the palm.

The local parameters are the metacarpophalangeal MCP(see Fig1(a)) X-rotations of the hand fingers except the thumb. The results of the hand model tracking in a synthetic video are shown in table1. The same table

illustrates the outputs of the one-step and the two-step iterative algorithms. We can see in table 1 that our dissimilarity function(d_F) provides the best results when it is compared with other well-known dissimilarity functions like the directed chamfer distance(d_C)[Bor88], the directed hausdorff distance (d_H)[HKKR93] and the non-overlapping function(d_{NOS})[OH99].

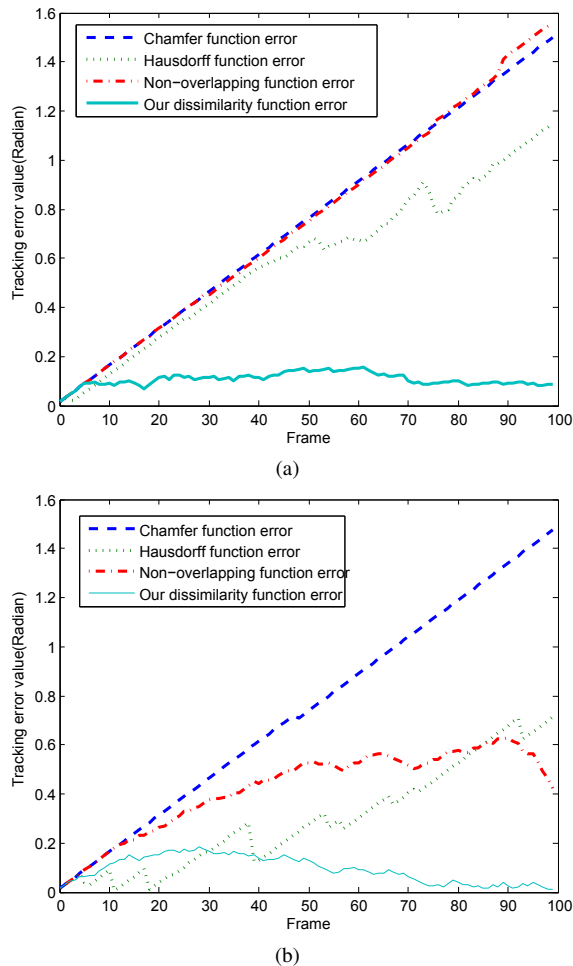


Figure 4: Z-Rotation tracking error.(a) Error obtained by the one-step iterative algorithm (b) Error obtained by the two-step iterative algorithm

To estimate the error of the tracking algorithm, we compute a difference between the tracking results and the ground truth. The tracking error is then plotted as a curve in Fig.4, in which we only consider the Z-Rotation. The curves represented in Fig.4 show that our dissimilarity function is more efficient than other functions compared with. Indeed, using our dissimilarity function the average tracking error is about 0.09 radian, while this error can exceed 1 radian with the others functions such as the directed chamfer distance. The same figure shows also that the two-step algorithm yields more robust results than the one-step algorithm. Specifically, in the case of the non-overlapping surface,

the two-step algorithm clearly improves the tracking performance. Indeed the average tracking error drop 0.76 radian with the one-step algorithm to 0.42 radian with the two-step algorithm. Our observations concerning the Z-Rotation tracking error could be still valid for other global and local parameters.

The processing is performed using a PC Intel-Centrino with duo-core processor and Nvidia graphic card (GeoForce 8600MGT), which is used to compute the model projection. For this purpose, we exploit the graphic library OpenGL² to obtain the model images. Besides robustness, we argue that the two-step iterative algorithm is faster than the one-step iterative one. Indeed, for synthetic video sequence, the processing rate is approximately about 11 frames per second for the two-step iterative algorithm, and 8 frames per second for the one-step iterative algorithm running on the same machine.

Another video sequence of four hundred frames of 320x240 pixels is processed using our algorithm. In this video sequence we track all the DOF of the hand except ones of the thumb. Indeed, the thumb is fixed and thus 18 parameters are estimated. For the six global parameters, we set experimentally the iteration number parameter of the downhill simplex to 10. This parameter is also set experimentally to 5 in order to estimate the local motion of each finger. Fig.5 shows the robustness of the two-step tracking algorithm compared with the one-step tracking algorithm. Indeed, the fingers motion tracking is more accurate for the two-step algorithm especially in the last frames of the video sequence. In fact, we can highlight that our two-step algorithm is more efficient to track some complex finger motion than the one-step algorithm, as we can see in the last frames of the video sequence shown in Fig.5. We can note also that the two-step algorithm is more powerful for tracking much DOF of the hand. Indeed, the processing rate for the video sequence (Fig.5) is about 5 frames per second for our two-step algorithm, and 2 frames per second for the one-step algorithm.

We evaluate also our algorithm for tracking more complex hand motions. As we can see in Fig.6 the proposed algorithm cannot deal with the self-occlusion problem since the hand tracking is lost after a few tens of images. Indeed the hand motion becomes very complicate to track when the x or y rotation of the palm becomes important. This problem is incontrovertible for monocular hand tracking. Wang and Provic[WP09] have tried to deal with the self-occlusion problem using a color glove. Their color glove allows to differentiate the palm down pose from the palm up one. They used a view-based approach but they could not estimate exactly the depth position of the palm. The self-occlusion problem still unresolved for monocular hand tracking.

² www.opengl.org

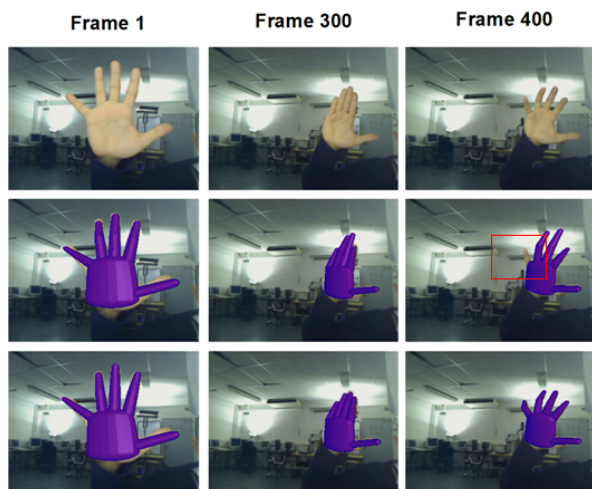


Figure 5: Tracking results for real images. The top row shows the selected frames of the video sequence, the middle row shows the tracking results of the one-step algorithm and the bottom row shows the tracking results of the two-step algorithm. The tracking loss is framed in red.

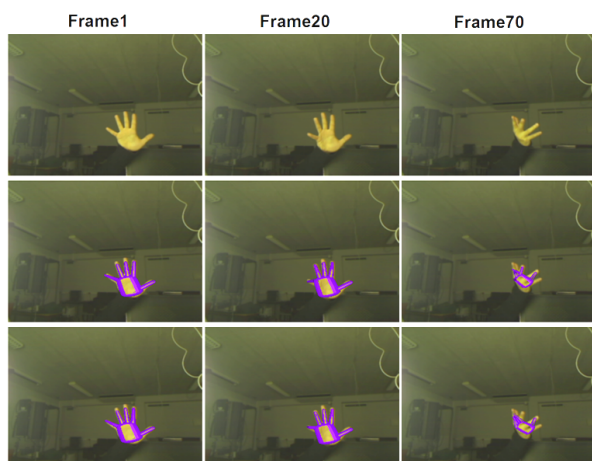


Figure 6: Self-occlusion problem illustrated with a video sequence of 70 frames

6 CONCLUSION AND FUTURE WORK

The main obstacle to articulated hand motion tracking is the large number of degrees of freedom (around 26) to be recovered. Search algorithms, either deterministic or stochastic, fall foul of exponential computational complexity. Deterministic approaches formulate the hand tracking problem as a minimization one, in which a dissimilarity function comparing a hand pose with a 3D hand model is to be minimized. The high number of the DOF compound the main problem of minimization algorithms: local minima. To overcome this problem, we present first a new dissimilarity function which

gives better results when it is compared with other well-known functions, like the directed chamfer or hausdorff distances as shown in the experimental results section. To minimize this dissimilarity function we propose a two-step algorithm operating on two stages. The first one gives the global DOF of the hand, i.e. position and orientation of the palm, whereas the second step provides the local DOF of the hand, i.e. finger joint angles. Operating in two stages, the proposed two-step algorithm reduces the complexity of the minimization problem. Indeed, it seems more robust to local minima than a one-step algorithm and improves the computing time needed to get the desired solution as shown in the experimental results section.

When using a single camera it remains difficult to deal with complex motion such as X-rotation or Y-rotation. Indeed, we are confronted with the problem of self-occlusion. To solve this problem, we plan to spend several cameras. Using multiple cameras, we will be able to deal with more complex hand motion.

A study in deepening course reveals that a GPU implementation of our method will significantly increase its computational power.

7 ACKNOWLEDGMENTS

Our research works have been achieved as part of the PlayALL project, which aims to develop a cross-platform toolchain standard for entertainment and video games. We thank very much the Cap Digital Cluster and the General Directorate for Enterprises (DGE) for their financial support. We also thank Brice Michoud for its very helpful comments.

REFERENCES

- [BKMM*04] BRAY M., KOLLER-MEIER E., MUELLER P., GOOL L. V., SCHRAUDOLPH N. N.: 3d hand tracking by rapid stochastic gradient descent using a skinning model. In *1st European Conference on Visual Media Production (CVMP)* (March 2004), Chambers A., Hilton A., (Eds.), pp. 59–68.
- [BKMSG04] BRAY M., KOLLER-MEIER E., SCHRAUDOLPH N. N., GOOL L. V.: Stochastic meta-descent for tracking articulated structures. In *IEEE Workshop on Articulated and Nonrigid Motion, Conference on Computer Vision and Pattern Recognition (CVPR)* (Washington, DC, USA, 2004), IEEE Computer Society, p. 7.
- [Bor88] BORGEFORS G.: Hierarchical chamfer matching: A parametric edge matching algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 10, 6 (1988), 849–865.
- [DF98] DELAMARRE Q., FAUGERAS O.: Finding pose of hand in video images: a stereo-based approach. In *IEEE Proc. of the third International Conference on Automatic Face and Gesture Recognition* (1998), IEEE Computer Society, pp. 585–590.
- [HH96] HEAP T., HOGG D.: Towards 3d hand tracking using a deformable model. In *Face and Gesture Recognition* (1996), pp. 140–145.

- [HKKR93] HUTTENLOCHER D. P., KLANDERMAN G. A., KL G. A., RUCKLIDGE W. J.: Comparing images using the hausdorff distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 15 (1993), 850–863.
- [IB98] ISARD M., BLAKE A.: Condensation conditional density propagation for visual tracking. *International Journal of Computer Vision* 29 (1998), 5–28.
- [IKS07] IKE T., KISHIKAWA N., STENGER B.: A real-time hand gesture interface implemented on a multi-core processor. In *MVA* (2007), pp. 9–12.
- [KX06] KATO M., XU G.: Occlusion-free hand motion tracking by multiple cameras and particle filtering with prediction. *IJCSNS International Journal of Computer Science and Network Security* 6, 10 (2006), 58–65.
- [LWH00] LIN J., WU Y., HUANG T.: Modeling the constraints of human hand motion. In *HUMO '00: Proceedings of the Workshop on Human Motion (HUMO'00)* (Washington, DC, USA, 2000), IEEE Computer Society, p. 121.
- [NM65] NELDER J. A., MEAD R.: A simplex method for function minimization. *The Computer Journal* 7, 4 (January 1965), 308–313.
- [OH99] OUHADDI H., HORAIN P.: 3d hand gesture tracking by model registration. In *Proc.IWSNHC3DI99* (1999), pp. 70–73.
- [RASS01] ROSALES R., ATHITSOS V., SIGAL L., SCLAROFF S.: 3d hand pose reconstruction using specialized mappings. In *ICCV* (2001), pp. 378–385.
- [RK94] REHG J. M., KANADE T.: Visual tracking of high dof articulated structures: An application to human hand tracking. In *European Conference on Computer Vision* (1994), Springer-Verlag, pp. 35–46.
- [RK95] REHG J. M., KANADE T.: Model-based tracking of self-occluding articulated objects. In *ICCV* (1995), pp. 612–617.
- [SKS01] SHIMADA N., KIMURA K., SHIRAI Y.: Real-time 3-d hand posture estimation based on 2-d appearance retrieval using monocular camera. In *RATFG-RTS '01: Proceedings of the IEEE ICCV Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems (RATFG-RTS'01)* (Washington, DC, USA, 2001), IEEE Computer Society, p. 23.
- [SMC01a] STENGER B., MENDONCA P. R. S., CIPOLLA R.: Model-based 3d tracking of an articulated hand. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on* 2 (2001), 310.
- [SMC01b] STENGER B., MENDONCA P. R. S., CIPOLLA R.: Model-based hand tracking using an unscented kalman filter. In *Proc. British Machine Vision Conference, volume I* (2001), pp. 63–72.
- [Ste04] STENGER B. D. R.: *Model-Based Hand Tracking Using A Hierarchical Bayesian Filter*. PhD thesis, University of Cambridge, 2004.
- [Tos06] TOSAS M.: *Visual Articulated hand tracking for Interactive Surfaces*. PhD thesis, University of Nottingham, 2006.
- [WH99] WU Y., HUANG T. S.: Vision-based gesture recognition: A review. In *GW '99: Proceedings of the International Gesture Workshop on Gesture-Based Communication in Human-Computer Interaction* (London, UK, 1999), Springer-Verlag, pp. 103–115.
- [WLH01] WU Y., LIN J. Y., HUANG T. S.: Capturing natural hand articulation. In *ICCV* (2001), pp. 426–432.
- [WLH05] WU Y., LIN J., HUANG T. S.: Analyzing and capturing articulated hand motion in image sequences. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 27, 12 (2005), 1910–1922.
- [WP09] WANG R. Y., POPOVIĆ J.: Real-time hand-tracking with a color glove. In *SIGGRAPH '09: ACM SIGGRAPH 2009 papers* (New York, NY, USA, 2009), ACM, pp. 1–8.