ORIGINAL ARTICLE

Ronan Boulic · Javier Varona · Luis Unzueta
Manuel Peinado · Angel Suescun · Francisco Perales

Evaluation of on-line analytic and numeric inverse kinematics approaches driven by partial vision input

Abstract Despite its central role in the constitution of a truly enactive interface, 3D interaction through human full body movement has been hindered by a number of technological and algorithmic factors. Let us mention the cumbersome magnetic equipments, or the underdetermined data set provided by less invasive video-based approaches. In the present paper, we explore the recovery of the full body posture of a standing subject in front of a stereo camera system. The 3D position of the hands, the head and the center of the trunk segment are extracted in real-time and provided to the body posture recovery algorithmic layer. We focus on the comparison between numeric and analytic inverse kinematics approaches in terms of performances and overall quality of the reconstructed body posture. Algorithmic issues arise from the very partial and noisy input and the singularity of the human standing posture. Despite stability concerns, results confirm the pertinence of this approach in this demanding context.

R. Boulic (✉)
Virtual Reality Laboratory, Ecole Polytechnique Fédérale de Lausanne, Station 14, 1015 Lausanne, Switzerland
E-mail: Ronan.Boulic@epfl.ch
Tel.: +41-21-6935246
Fax: +41-21-6935328

J. Varona · F. Perales
Dept Mat. i Informatica, Universitat de les Illes Balears (UIB), Balears, Spain
E-mail: vdmijvg4@uib.es
E-mail: Paco.Perales@uib.es

L. Unzueta · A. Suescun
CEIT and Tecnun (University of Navarra), San Sebastian, Spain
E-mail: lunzueta@ceit.es
E-mail: asuescun@ceit.es

M. Peinado
Escuela Politécnica University of Alcalá, Alcalá, Spain
E-mail: manupg@aut.uah.es

Abbreviations IK: Inverse kinematics · PIK: Prioritized inverse kinematics · dof: Degree of freedom

1 Introduction

The sense of movement has been under-exploited until now in classical interfaces. Integrating the kinaesthetic sense at a larger scale than desktop manipulations is fundamental for building effective enactive interfaces where our dexterity and full body postural knowledge can be exploited. Exploiting the sole 3D location of one or two hands is indeed not sufficient for the evaluation of complex tasks in virtual environments. For example, very often a new product to assess is part of a cluttered environment hence raising accessibility issues for the human operator in charge of using or maintaining it. Therefore, it is crucial to be able to easily specify the full body posture of a virtual mannequin for conducting such evaluations on the virtual prototype as early as possible in the conception process. The present paper targets such objective with the long-term aim of offering a real-time non-invasive technology for the intuitive specification of human full body postures while interacting with complex virtual environments. Until now, the exploitation of real-time motion capture of full body human movements has been limited to niche applications such as the expressive animation of a virtual character in a live show (Sturman 1998). Multiple factors hinder a wider adoption of full body movement as a popular 3D user interface. Among others, we can cite: the invasiveness of the sensor system, the limited acquisition space and sensor precision, the spatial distortions, the high dimension of the posture space, and the modeling approximations in the mechanical model of the human body. These sources of errors accumulate and result in an approximate posture. It can be sufficient for performance animation where expression counts the most. However, if precise spatial control is desired, this channel may not suited for evaluating complex interaction with virtual objects.

The factor we want to improve in the present study is the comfort of the user through a non-invasive vision-based acquisition technology. By means of vision-based inputs, we recover the user's full body posture while he/she interacts in front of a workbench. From the images of two standard color cameras, we automatically locate the 3D positions of the hands and the head. In addition to these 3D positions, we can also exploit an estimate of the center of the trunk segment when necessary. Then, we compare the performances of two inverse kinematics (IK) algorithms for the posture recovery of the full body. It should be noted that this context is very challenging, as the problem is highly under-determined and the standing human posture is often close to the well-known postural singularity with fully extended arms and legs. The comparison focuses on the computing cost, as we want to ensure proper integration of the user in the interaction loop. A slightly less important criterion is the quality of the spatial correspondence between the real hands and the virtual hands resulting from IK reconstructed postures. It has been visually assessed while performing various reach movements. Conversely, we strongly feel that whenever possible, it is important to ensure that the reconstructed posture is naturally balanced. For this reason, we enforce a balance constraint with the numeric prioritized inverse kinematics (PIK) approach. The correctness of the resulting balance behavior is confirmed with a small on the fly collision detection experiment.

The paper is organized as follows. Section 2 recalls the background in real-time full-body motion capture, especially for 3D interactions. In the remainder of the paper, we first provide an overview of the system prior to detail its vision and movement recovery components (Sects. 3–6). Test cases comparing the two IK approaches follow in Sect. 7 and we conclude in Sect. 8.

2 Background

Real-time full-body motion capture has a long history in performance animation (Sturman 1998). The very limited user-friendliness of the exoskeleton technology used at that time has prevented its wider adoption. The magnetic sensor technology has started to be used in the 1990s by Badler, who investigated the use of four magnetic sensors (waist, head and both hands) for driving the posture of a human model with IK (Badler et al. 1993). The goal is to recreate human postures while minimally encumbering the end user with sensor attachments. However, the uncontrolled degrees of freedom (dofs), like the swivel angle of the arms, can lead over time to important differences between the end user and the virtual human model. Molet has described an approach suppressing this ambiguity by using more sensors (Molet et al. 1999). Other approaches, identifying also the skeleton structure and segment lengths, were proposed by Bodenheimer et al. (1997) and O'Brien

et al. (2000). Recent works show a renewed interest to propose less invasive approach exploiting a reduced number of sensors (Grochow et al. 2004; Chai and Hodgins 2005). These approaches first learn local models of human motions from a predefined motion capture database. Then, using these local models, the vision inputs from retro-reflective markers guide the motion recovery. However, presently these techniques are not used for 3D interaction, mainly due to the fact that user's motions are restricted to the previously learned movements. Another possibility is to compensate the missing information through constraints. The possibility to associate a strict priority to a constraint is a key aspect for the success of such an approach, as highlighted in the context of interactive posture optimization (Baerlocher and Boulic 2004). Alternatively, analytic IK approach is generally more efficient in terms of computing cost but it does not allow assigning priority levels to the constraints (Tolani et al. 2000).

Reconstructing the human motion from video image analysis has received a great attention in Computer Vision (Wang et al. 2003; Moeslund and Granum 2001). However, most of the current approaches are not real-time, hence making difficult the comparison of our approach with non-real-time ones (Horain and Bomb 2002). For our objective, the real-time constraint is very important due to our goal of using the captured postures as an input for a perceptual user interface in virtual environments. An interesting prior work in real-time is the one of Wren et al. (2000) at the M.I.T. MediaLab. In this work, the authors present a system for the 3D tracking of the upper human body in front of a virtual reality device. No performance evaluation of their system is given, however. Moreover, the possible gestures are restricted to a set of predefined movements learnt previously. This approximation reduces the searchable space of human motions by learning-from-example. Our system isn't restricted to a set of predefined movements and the delay between the user's movement and the system response is not noticeable by the user. Finally, we wish to point out that our work includes the trunk joints compared to other work dedicated to upper body tracking (Jojic et al. 1999). This allows us to include a balance constraint in our system to ensure the static equilibrium of the user's skeleton posture.

3 System architecture

For recovering body movements, the user is located in an interactive space that consists of a workbench with two projection screens. This space is instrumented with a stereo camera pair. The stereo pair is used to capture the motions of certain parts of the user's body. The choice of which parts of the body depends on the particular body posture recovery algorithm. This configuration allows the user to view virtual environments while standing in front of the workbench. Gesture and

manipulation occur in the workspace defined by the screens and the user (Fig. 1). The workspace requirements are:

- The background wall covered with chroma-key material. The system may work without chroma-key background; however, using it ensures a real-time response.
- Only one person shall be present in the space.
- The color of the user's clothes should not be similar to skin color.
- The skin colored body parts, other than the hands and face, shall not be visible. For example, the user should not roll up his sleeves.

The images from the two synchronized color cameras will be the inputs of the system, if the above requirements are met. Usually, locating all the user body joints in order to recover the posture is not possible with computer vision algorithms only. This is mainly due to the fact that most of the joints are occluded by clothes. IK approaches can solve the body posture from their 3D position if we can clearly locate visible body parts such as face and hands. We propose a scheme where these visible body parts (hereafter referred to as *end-effectors*) are automatically located in real-time and fed into an IK module, which in turn can provide a 3D feedback to the vision system (Fig. 2). Two types of IK algorithms, one analytic and one numeric, have been integrated in the system for the purpose of comparing their relative performance for real-time body posture recovery.
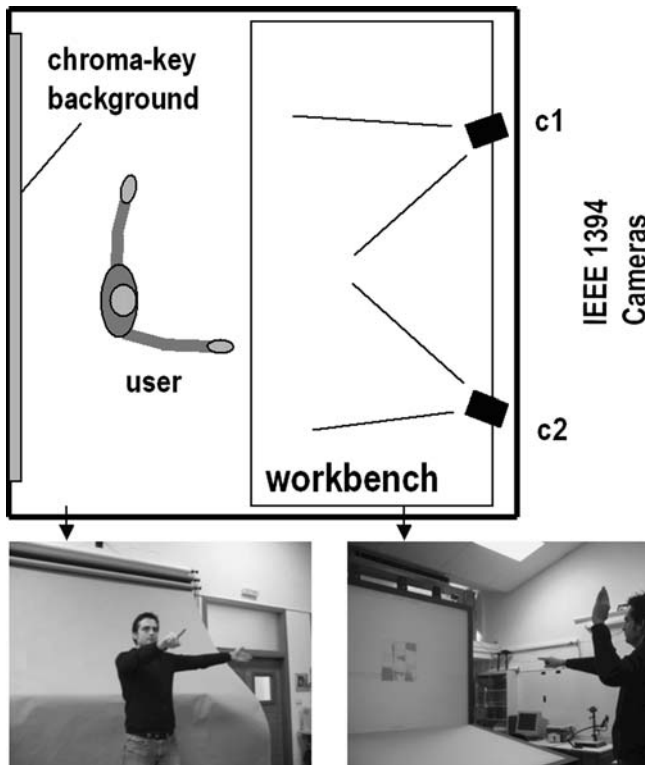

Fig. 1 Vision system layout

4 The vision system

We apply chroma-keying, skin-color segmentation and 2D-tracking algorithms for each image of the stereo pair to locate the user's end-effectors in the scene. Then, we combine this result in a 3D-tracking algorithm to robustly estimate their 3D positions in the scene. Figure 3 shows this process schematically.

First, a mask is built with the image pixels belonging to the user by applying chroma-keying techniques to the original images in the Human Segmentation process. This mask is used to constraint the skin-color segmentation algorithm to those pixels belonging to the user's silhouette. Our skin-color detection algorithm finds in real-time the skin-color pixels present in the image. The results of this skin-color detection are skin-color blobs, which are the inputs of our 2D-tracking algorithm. This algorithm labels the blobs pixels using a hypothesis set from previous frames (Varona et al. 2005). The 2D-tracking results are feed to the 3D-tracking algorithm to robustly obtain an estimation of the end-effectors 3D positions.

The 3D-tracking algorithm uses a Kalman filter to estimate the end-effector 3D position from the image measurements obtained by the 2D-tracking algorithm. The use of a Kalman filter ensures a robust estimate for the end-effectors and smoothens the estimations between consecutive frames for minimizing the end-effectors positional jitter that may cause wobbles on the full posture estimation. Besides, we use the predictions from the Kalman filter to establish the correct correspondence between each end-effectors 2D position. In order to do this, we first triangulate all possible combinations of 2D measurements from the two images to obtain the 3D position candidates of each end-effector. Then, for each end-effector, we select the candidate nearest to the position predicted by the estimation filter. Figure 4 shows the results of this process by back projecting the cor-
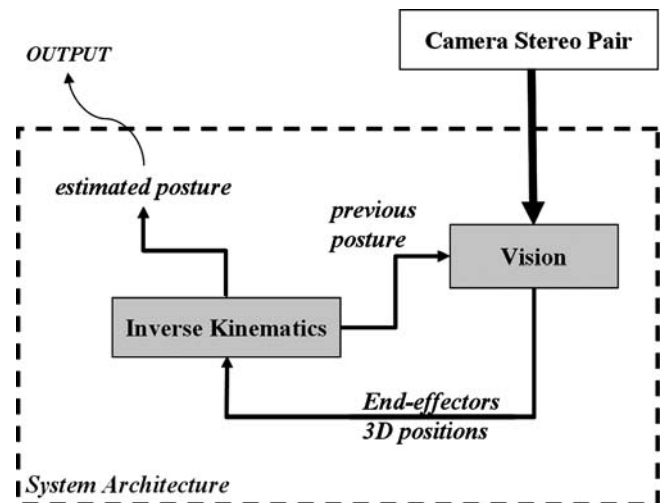

Fig. 2 General architecture of the system

Fig. 3 Vision process

rected associate end-effectors 3D position in the 2D images of the stereo pair.

As explained before, the data required by the IK algorithms are the end-effectors' goal position. In our case, the end-effectors are the 3D positions of the wrists joints. In order to locate the wrists from the hands positions, we use the 2D ellipses found by the 2D-tracking algorithm, the 3D hand positions from the 3D-tracking algorithm, and the previous 3D positions of elbows estimated by the IK algorithm. This is done by searching in the image for the intersection between a 2D line, as defined by the back projection of the corresponding elbow and 3D hand positions, and the 2D ellipse. Then, from the 2D wrist positions, we compute their 3D coordinates by triangulation. Two examples of wrist location determined in this way can be seen in Fig. 5.

In addition, an optional algorithm can estimate the location of the user's center of mass as indicated in Fig. 3. This algorithm computes the image moments up to order 1 of the user's binary silhouette:

$$m_{ij} = \sum_x \sum_y x^i y^j I(x,y), \tag{1}$$

where $I(x,y)$ is the pixel value at the (x,y) location (1 if the pixel belongs to the binary silhouette or 0, if not). These values are used to find the center of gravity of the human shape in the image:

$$c_x = \frac{m_{10}}{m_{00}}, \quad c_y = \frac{m_{01}}{m_{00}}. \tag{2}$$

We triangulate both centers of gravity to obtain an approximation of the user's 3D center of gravity.

Figure 6 displays two typical support configurations where the weight distribution on both feet is either equal (Fig. 6a) or privileges one foot (Fig. 6b). In both cases, the 3D estimation resulting from the two silhouettes' center of gravity consistently projects over the support area.

The body segments' length is identified through the manual annotation of six joint centers, namely the shoulders, the elbows and the wrists, on both images from the stereo pair. This is made only once at the calibration stage for which the user adopts a symmetric standing-up posture with the arms resting along the body. The joints' 3D position is deduced by triangulation. They are subsequently exploited in both IK approaches to initialize their respective simplified skeletons, as detailed in the next sections.

5 Analytic inverse kinematics

The analytic IK first controls the global position and orientation of a trunk segment and, in a second stage, the joint state of two arms where each arm is composed of a shoulder (3 dofs) and an elbow (1 dof). The controlled effectors being the wrist joint centers, the local mobility of these joints is not exploited in the present study.

5.1 Trunk control

The trunk control is necessary even if the user is interacting only through arms gestures. Indeed the trunk

Fig. 4 Hands and face tracking

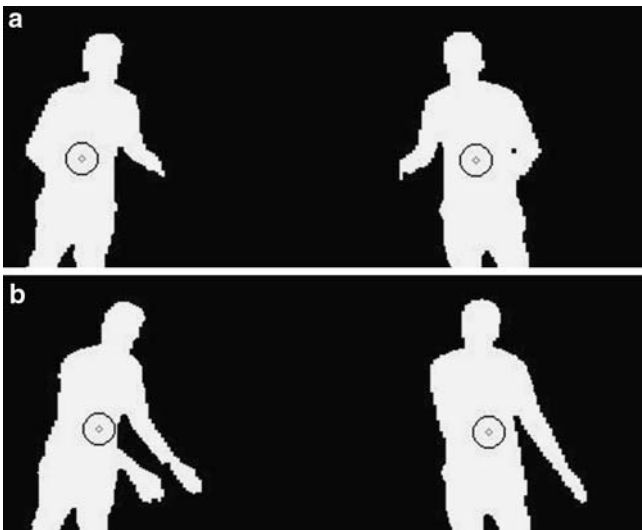**Fig. 5** 3D position of the hands end-effectors

contributes both to reach actions and to preserve the balance of the whole body by bending and twisting. For this reason, the algorithm first estimates the main axes of the trunk from information delivered by the vision system (Fig. 7a):

Longitudinal axis L normalized vector difference between the position of the head and the one of the vision-based estimated center of gravity.

Transversal axis T first we compute the vector linking the hands and we retain its horizontal component, noted H. The heuristic of computing a weighted mean between H and the body lateral axis is proven to give good results for the tests that were conducted in the present study.

The trunk orienting and positioning proceeds as shown in Fig. 7b. First, the trunk longitudinal axis is aligned with the axis L by acting on the "humanoid root" rotation (Fig. 7b, left). Second, the trunk is rotated along its longitudinal axis so that the trunk lateral axis is aligned with the axis T (Fig. 7b, middle). Third, the whole body is translated so that the head center is brought on the 3D position identified by the vision system (Fig. 7b, right).

5.2 Arm posture determination

The second stage of the analytic IK algorithm is to determine the arms' posture from the updated position of the shoulder joints and the 3D wrist positions obtained with the vision analysis. This is organized in three stages as illustrated for one arm in Fig. 8:

- First, the vector linking the shoulder center, noted A, to the wrist center, noted C, is aligned with the vector AD where D is the goal location for the wrist (Fig. 8a, left).
- Second, both the shoulder and the elbow are adjusted so that C slides toward D along AD (Fig. 8a, right).

Fig. 6 Two stereo pairs highlighting the center of the silhouettes used for the estimation of the human center of gravity; **a** with equal weight distribution, **b** one foot supports more weight than the other
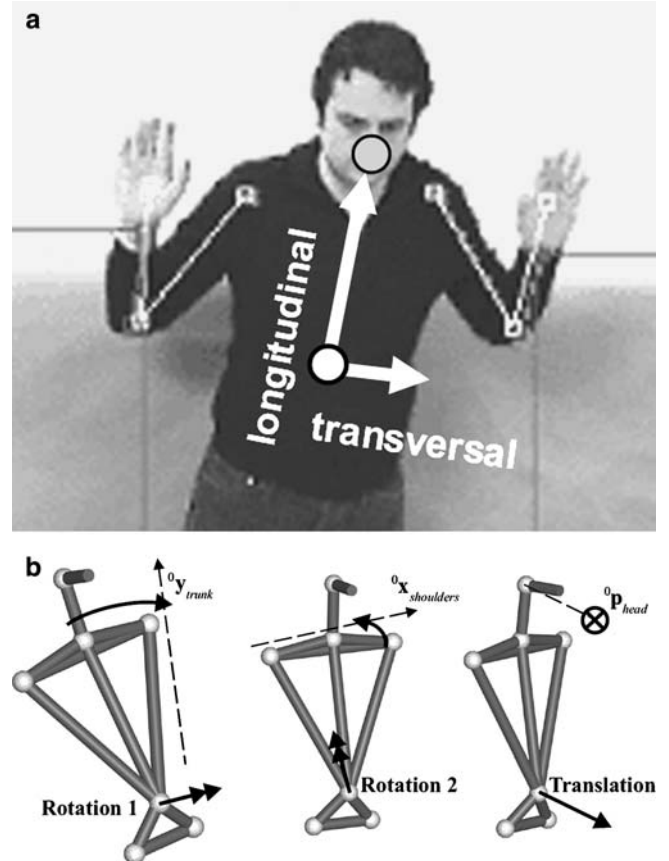
Fig. 7 The three stages of the arm posture control bringing the wrist C on the goal position C′ (stages 1 and 2) followed by the swivel angle optimization (stage 3)

Fig. 8 Definition of the longitudinal and the transversal axes from the estimation of the silhouette center of gravity (*white*) and the head position
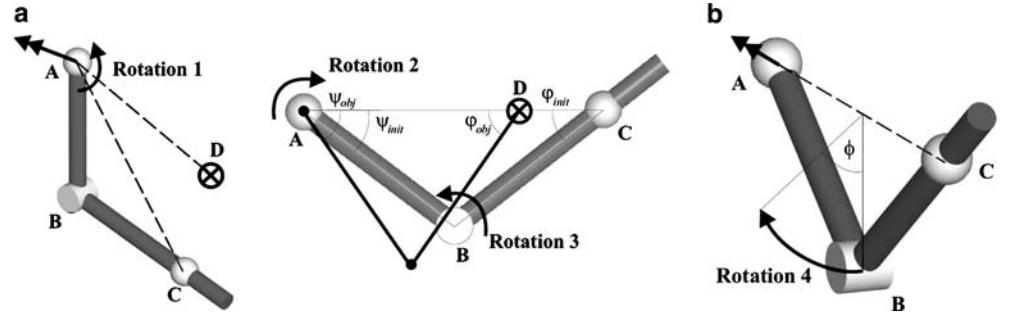
- Third, the remaining dof is called the swivel angle (Fig. 8b). The retained value for the present study minimizes an additional cost function that attracts the shoulder Euler angles toward their mid-range value. The cost function is evaluated for sampled values of the swivel angle, first every 0.1 rd, then, for the best candidate angle S a finer sampling of 0.01 rd is used to make the final selection of the swivel angle within $[S-0.1$ rd, $S + 0.1$ rd$]$. This simple approach is much faster than a previous approach based on genetic algorithm (Unzueta et al. 2005).

6 Numeric prioritized inverse kinematics

The major difference with an analytic approach is the dependency on a first order approximation requiring to perform small variations to converge toward the goal positions; this is achieved usually in more than one iteration. In addition, the order of complexity of numeric PIK is both quadratic with the number of dofs and linear with the constraints dimension. A clever analysis of the computing cost, together with the exploitation of an algorithm enforcing strict priority levels among constraints, help to reduce its impact (Baerlocher and Boulic 2004). However, these factors are still determinant when designing an articulated model and its associated set of prioritized constraints for real-time use.

6.1 The simplified articulated body model

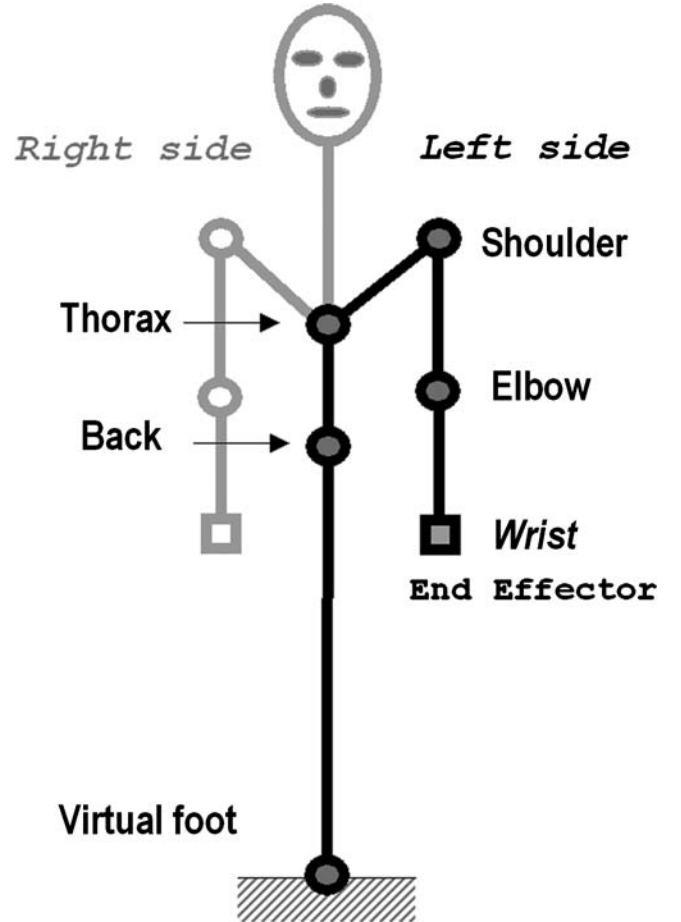The present study exploits an articulated body model with 15 dofs distributed as follow (Fig. 9):

Virtual foot (2 dofs) roots the body to the floor with frontal and lateral axes of rotation.

Back (2 dofs) corresponds to the beginning of the spine with frontal and lateral axes of rotation.

Thorax (3 dofs) all rotation axes.

Shoulders (2 × 3 dofs) all rotation axes.

Elbows (2 × 1 dof) only the flexion-extension.

We use the initial joint positions of the shoulders, the elbows and the wrists for computing the segments' lengths that remain constant for the rest of the session. We can derive the location of the other joints as the relative proportion of the lower body segment and the back segment are considered constant (see Fig. 9).

The wrists are end-effectors controlled by the PIK algorithm. Each of them exploits the mobility of the three joints modeling the leg and the trunk. Besides, joints are assigned some joint limits to prevent unnatural

Fig. 9 Simplified body model exploited by the numeric inverse kinematics. The kinematic chain involved in controlling the left wrist end-effector is highlighted in *black*. The leg and trunk joints are also used by the right wrist

posture to appear; in particular the shoulder and elbow joint limits are critical to prevent self-collision or the fully flexed singular posture to occur. The key motivation for modeling the simplified leg and trunk is to associate an approximate mass distribution so that the PIK algorithm can simultaneously ensure the balance of the body by controlling the position of its center of mass (Boulic et al. 1996). More precisely, only the two horizontal dimensions are constrained to keep their initial value computed at the calibration stage (Fig. 9). As a consequence, the center of mass is free to move up and down but only on the vertical line passing through the foot. This constraint is given a high priority, as it its enforcement is important for the plausibility of the resulting posture. Assigning the highest priority to this constraint means that it is enforced before the others within only a very small number of convergence steps. Therefore, this choice guarantees that most of the time the posture is balanced while converging to enforce the low priority constraints. For the present study, the mass is distributed as follows: 50% is attached to the virtual foot joint roughly at the height of the mid-thigh, 25% is attached to the back joint at mid-distance to the thorax joint, the final 25% is attached to the thorax joint at mid-shoulders distance.

6.2 Overview of the prioritized inverse kinematics

The multiple priority IK (also called prioritized IK, or PIK) is exploited for reconstructing an anatomically correct posture of the user (i.e., its joint state, θ) from the 3D location of selected end-effectors (noted x) measured with the vision system and used to constrain the posture. We give here a general overview of the method while Sect. 6.4 describe the specific set of constraints and priorities exploited in this study.

Our general architecture is based on the linearization of the set of equations expressing Cartesian constraints x as functions of the joints' dofs θ. We denote J the Jacobian matrix gathering the partial derivatives $dx/d\theta$. We use its pseudo-inverse, noted J^+, to build the projection operators on the kernel of J, noted $N(J)$. Our approach relies on an efficient computation of projection operators allowing to split the constraints set into multiple constraint subsets associated with an individual strict priority level (Baerlocher and Boulic 2004). The provided solution guarantees that a constraint associated with a high priority is achieved as much as possible while a low priority constraint is optimized only on the reduced solution space that does not disturb all higher priority constraints. For example, such architecture is particularly suited for the off-line evaluation of reachable space by a virtual worker; in such a context the balance constraint is given the highest priority while gaze and reach constraints have lower priority levels (Boulic et al. 2004).

Figure 10 provides an overview of our PIK control. The outer convergence loop is necessary as the linearization is valid only within the neighborhood of the current state; such a small validity domain requires to limit the norm of any desired constraint variation Δx toward their respective goal to a maximum value and to iterate the computation of the prioritized solution $\Delta\theta$ until the constraints are met or until the sum of the errors reaches a constant value. The present study exploits the PIK algorithm with various maximum numbers of convergence steps per vision-based data sample. Figure 10 also highlights the clamping loop handling the inequality constraints associated to the mechanical joint limits. Basically, we check whether the computed prioritized solution $\Delta\theta$ leads to violate one or more joint limits. If it is the case, equality constraints are inserted to clamp the flagged joints on their limit and a new prioritized solution is searched in the reduced joint space (Baerlocher and Boulic 2004; Boulic et al. 2005).

6.3 Specific issues in the vision-driven real-time context

The classical stability-robustness tradeoff in inverse problems finds a clear illustration with our vision-driven IK case study. We have to deal with multiple sources of instability while trying to ensure that end-effectors reach their assigned goals sufficiently quickly. Let us review the key sources of potential instability or slow-down factors and how we can handle them in our PIK framework:
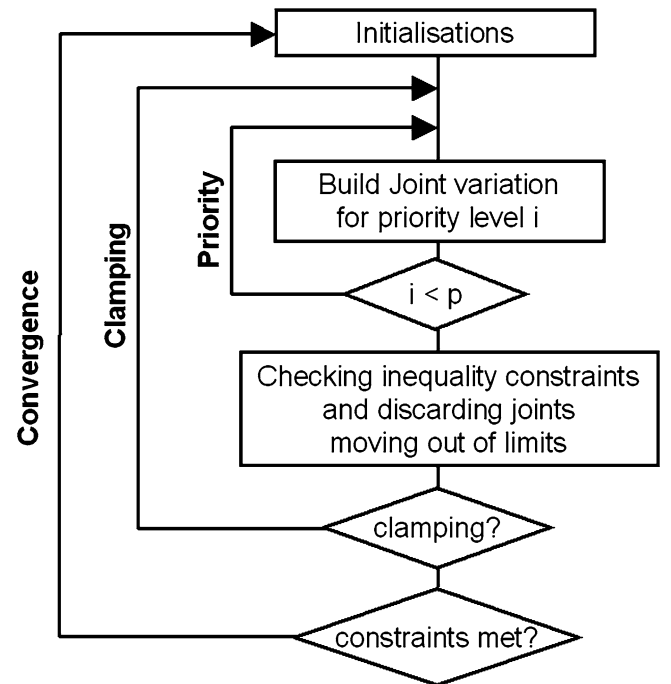
Fig. 10 The outer loop iterates the construction of the first order solution with priorities (inner loop) and joint limit enforcement (clamping loop)

Noisy input the low image resolution and the combination of various uncertainties result in jitters at the level of the 3D location of the position of the wrists' goal. To avoid this instability to be transmitted to the articulated structure, the vision module can filter the goals' location, or the IK module can increase the damping factor exploited to handle the singularities of the posture (Maciejewski 1990). In both scenarios, there is a risk of introducing a lag when the wrists move rapidly over large distances. We have experimented successfully a combination of filtering and damping.

Approximate skeleton the skeleton is built in the initial calibration posture (Fig. 9) where the approximate location of the back, shoulders, elbows, and wrists are provided by manually pointing on the stereo pair images. These 3D vectors are then provided to the skeleton initialization function, which infer the location of the virtual foot joint and the thorax joint from the distance between the back and the mid-shoulder location. This phase may introduce some errors in the arm segment lengths, which prevent the exact matching of the real skeleton with the model in other postures. Special care is required in the calibration phase.

Singular postures a posture is singular for a numeric IK algorithm when no solution can be computed for the desired goals. For example, when the wrist goal is too far and unreachable. More surprisingly, a fully extended arm is singular when the wrist goal is exactly on the line linking the wrist to the shoulder (Boulic et al. 2005; Maciejewski 1990). This is due to the first order approximation of the PIK algorithm. When a posture is precisely in a singular configuration, the homogeneous solution provided by the pseudo-inverse J^+ has a null norm. A stability problem appears in the neighborhood of a singular posture, as the associated Jacobian matrix J becomes ill-conditioned. This induces its pseudo-inverse to provide a solution, which norm can be extremely high. It can produce strong instabilities. The solution to this problem is to adopt a damped least square inverse (Maciejewski 1990; Baerlocher and Boulic 2004) and to tune the damping factor to a value removing the instability without introducing a lag. The fully flexed arm singular posture is prevented owing to the elbow flexion joint limit.

The fully extended arm posture this posture deserves a special mention in terms of source of troubles. The previous point has described how to remove the instability due to this singularity. It works fine when the wrist goal is too far. However, it tends to prevent the arm from flexing when the goal is on the line linking the wrist to the shoulder. As a consequence, it may slow down the convergence or worse result in other joints taking on the task of moving the wrist to its goal location. It is needless to say that the resulting posture, while possible, may not be very plausible. The problem is that the fully extended arm is a very naturally adopted posture; it is even in the calibration posture. We have analyzed this problem and proposed a solution through the concept of observers (Boulic et al. 2005). Basically, we detect the conjunction of the fully extended arm posture together with the occurrence of a wrist goal in the shoulder direction (within a tolerance). When this condition is met, we activate an elbow flexion increment within the low-level optimization task (cf. Sect. 6.4). This temporarily forces the arm to flex and speeds up the convergence.

First order approximation and conflicting constraints the PIK architecture theoretically ensures that conflicting constraints can exploit the same joints. However, this is valid only in the neighborhood of the current state, hence requesting the definition of this validity domain through thresholds enforced on the desired Δx. Improper threshold values is a common source of instability that requires some tuning.

Performances this work has demonstrated the feasibility of integrating a numeric PIK algorithm within a real-time motion capture loop. The present system has been implemented in Visual C++ using the OpenCV libraries (Bradski and Pisarevsky 2000) and it has been tested in a real-time interaction context on an AMD Athlon 2800 + 2.083 GHz under Windows XP. The images have been captured using two DFW-500 Sony cameras with an IEEE1394 connection. The cameras provide 320×240 images at a capture rate of 30 frames per second (more details are given in Table 2; Sect. 8).

6.4 Constraints hierarchy

The approach presented in the present paper relies on four levels of priorities ensuring not only the user-defined goals (wrist position) but also general property of the posture space such as the balance. Due to its great importance in the overall quality of the posture, the constraint enforcing the body balance is given the highest priority. This means that all other constraints of lower priority are found in the sub-space of the balanced postures. Therefore, it ensures that the intermediate postures show at least this quality before achieving all constraints. Such a channeling of the convergence has two positive consequences: first, this removes some class of local minima that would have otherwise occurred in an approach without priorities, and second, the intermediate postures, being balanced, are better accepted by the viewer even if all constraints are not met. This is important in a real-time context, as our time budget may allow for one or a few IK convergence steps only per vision-based input data.

The second rank constraint is the one attracting the wrist end-effectors toward their vision-driven goal position. For each wrist, all the joints from the elbow to the virtual foot root contribute to the achievement of the constraint. Figure 11 highlights how the two constraints balance and reach lead to a natural configuration of the whole chain. The third rank constraint is the one attracting the shoulders center of rotation toward their

initial location in the calibration standing posture. Finally, at the lowest priority level, we add an optimization vector expressed directly in the joint variation space [see

Fig. 11 The constraint associated with the center of mass ensures the balance of the whole model (numeric inverse kinematics solution)

Baerlocher and Boulic (2004) for details]. This vector is composed of two distinct components:

Minimization of the joint amplitude this corresponds to an attraction toward the initial standing posture. It resolves the ambiguity of the swivel angle by attracting the elbow along the trunk.

"Full extension" avoidance term for the elbow in case the condition described in Sect. 6.3 is met, a flexion term replaces the value proposed by the joint minimization for the elbow.

Table 1 summarizes the hierarchy of four priorities levels.

7 Results

We have first evaluated the PIK architecture on a simple test case: first, flexing the arm to the maximum flexion and extending back to the full extension, and second, raising one arm laterally up to the horizontal level and then trying to reach the furthest possible point. Despite its simplicity, the first test case highlighted the issues linked to the correct initialization of the segments' length and to the arm singularity described in Sect. 6.3. The observer concept proves to be well-adapted in the present real-time context as it correctly helps the arm to flex when the first order approximation is numerically "blind" to this second order solution. The second test case is illustrated more in detail in the following subsections. Performances are discussed in Sect. 8.

7.1 Far lateral reach with prioritized inverse kinematics

This test case underlines the interest of the center of mass control to maintain the balance of the body. Indeed, when the arm moves sideward to reach a distant point in space, the lower body moves in the opposite direction so that the center of mass still projects over the virtual foot (Fig. 11). As indicated in Sect. 6.1 the location of the simplified body center of mass can be computed from the knowledge of its current posture and of the (constant) mass distribution. The goal of the center of mass position constraint is invariant, as it is always attracted toward the vertical line passing through
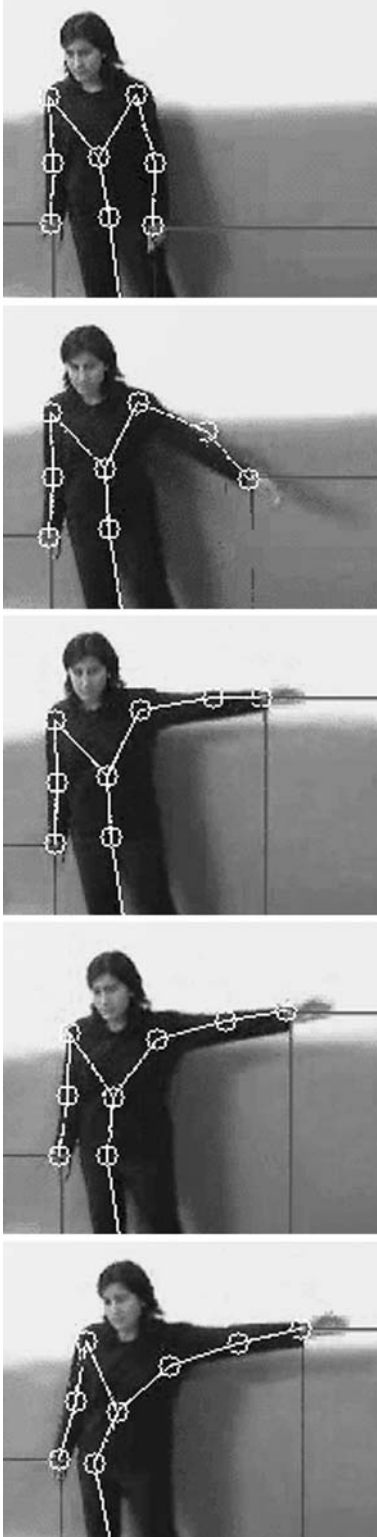
Table 1 Hierarchy of prioritized constraints

Constraint	Priority rank	Dimension
Projection of the center of mass over the virtual foot to maintain the balance	1	2
Wrists position control	2	$2 \times 3 = 6$
Shoulders position control	3	$2 \times 3 = 6$
Joint space optimization vector, minimizing the joint amplitude and conditionally avoiding the extended arm singularity	4	Joint space dimension

its initial position (the performer is not moving the feet in the present study). For these reasons, the numeric PIK does not need the estimated center of the silhouette, hence sparing some computing time.

## 7.2 Importance of the trunk postural control

This comparison highlights the interest of controlling the trunk posture even if the user is standing in front of the workbench. The left column of Fig. 12 displays both the numeric and the analytic IK solutions. The latter is computed without trunk control, which prevents the left wrist from reaching its goal in the bottom image (chain highlighted with the white squares and dotted lines). The numeric solution better reflects the user posture owing to the center of mass control that induces the natural counter-balanced posture. Figure 12 right column illustrates the complete analytic solution including the trunk orientation. The recovered movement is much more realistic, as the shoulder joints are now correctly located in space.

7.3 Handling an imaginary box

In this case study, the user acts as if he is carrying an imaginary box from side to side. The movement involves a complex deformation of the trunk (Fig. 13). Both approaches successfully recover the shoulder locations through their respective trunk postural control. On the other hand, the elbow location, determined by the swivel angle optimization, reflects the specific and somewhat arbitrary choice made by each approach.

We measured the average and the standard deviation of the position error on both wrists for this movement (Table 2). More precisely, we computed the norm of the position error between the current wrist position as determined with an IK method and the desired goal position determined by the vision system (considered as the ground truth data for the IK methods evaluation). The average error due to the analytic IK is 2.2 mm with a standard deviation of 10 mm (most of the time the error is very small except for a small number of distant goal positions). The error due to the PIK strongly depends on the number

Fig. 12 Analytic inverse kinematics solution (*chain with squares*). *Left*—only hand attraction. *Right*—exploiting the head and the center of mass information
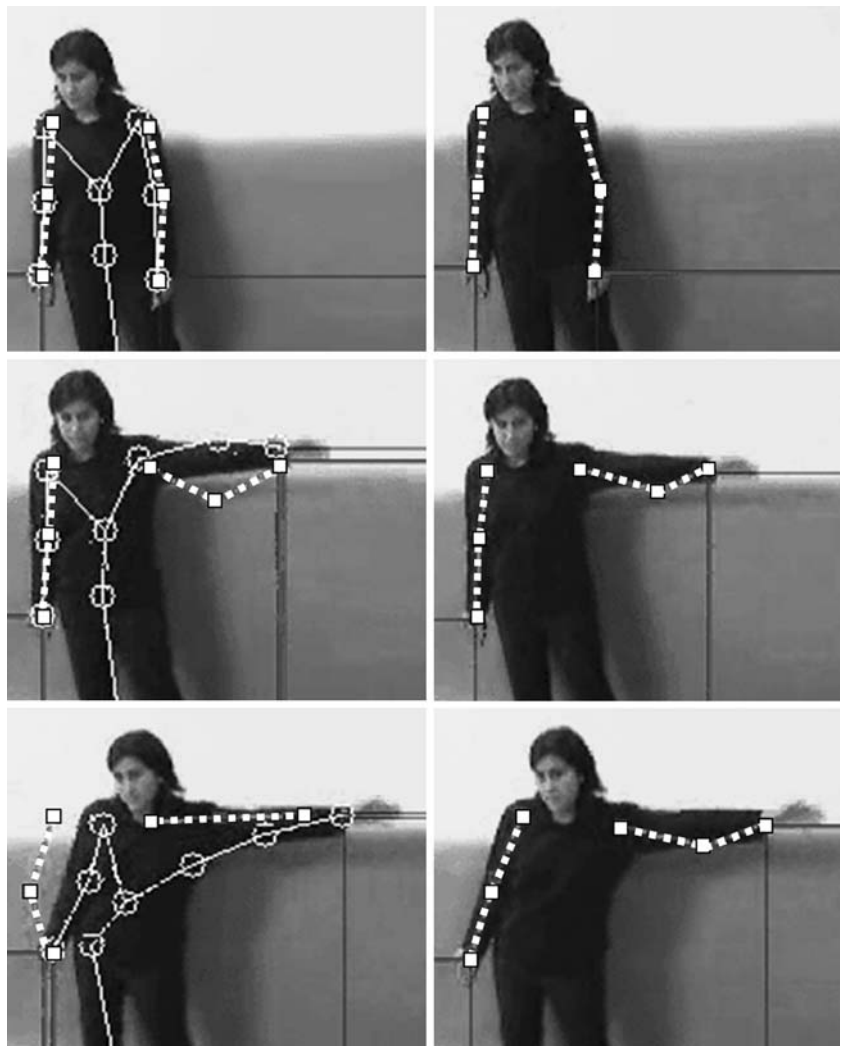
Fig. 13 Moving an imaginary box (real-time duration, 5.5 s). *Squares*—analytic solution, *circles*—numeric solution

of convergence steps. For this reason, it has been run multiple times on the same vision-based input but with an increasing number of convergence steps: 1, 5, 10,

Table 2 Average position error and standard deviation on both wrist positions and for both inverse kinematics (IK) approaches for the movement illustrated in Fig. 13

Case (+ nb of convergence steps)	RMSE (mm)	Std Dev (mm)
PIK + 1 right wrist	103.4	133.3
PIK + 1 left wrist	108.3	142.5
PIK + 5 right wrist	4.8	12.0
PIK + 5 left wrist	5.1	12.3
PIK + 10 right wrist	2.6	11.0
PIK + 10 left wrist	2.3	9.5
PIK + 20 right wrist	2.5	10.9
PIK + 20 left wrist	2.3	9.5
Analytic IK right wrist	2.2	10.9
Analytic IK left wrist	2.2	10.0

PIK prioritized inverse kinematics

20. The average error norm is around 100 mm for a unique convergence step but decreases rapidly (5 mm for five convergence steps) until a similar minimum of about 2.5 mm for 20 steps. The best compromise still compatible with real-time interaction is to run the PIK with five convergence steps per vision-based input data (see also Table 3).

7.4 Collision detection

One key interest of identifying the user body posture is the possibility to check whether the body enters in collision with elements from the virtual environment. Many collisions cannot be inferred from the sole information of the hand position or by exploiting only the user silhouette. To demonstrate the feasibility of this class of application, we have placed a virtual cubic object at a small distance behind the user. Then we asked the user to reach a location *in front of* him/her. Despite the opposite direction of the reach movement with respect to the obstacle location, a collision does occur generally, as the user moves the lower body backward to counterbalance the forward movement of the upper body (Fig. 14). This collision is detected on-the-fly and reported as a highlighted graphical representation of the virtual obstacle in the field of view of the user. This information tells the user that such forward reach is not possible due to the collision.

A second example illustrates a user performing a hand movement in the horizontal plane that induces a collision between the elbow and a virtual box located on the side of the user (Fig. 15). The highlighted feedback is determinant for the awareness of the collision, as other clues such as the silhouette are useless in that context (Fig. 16). One potential application is to explore and characterize the reachable space of the user by tagging each sampled position of the hand trajectory as either *reachable* or *unreachable* when a collision occurs. The gathered feedback can be stored in a cloud of 3D points that can be analyzed later on more in details for the evaluation of the virtual environment.

Table 3 Evaluation criteria for the comparison of the two inverse kinematics (IK) approaches

	Performance AMD Athlon 2800 + 2.083 GHz	Continuity	Flexibility
Analytic IK	Stable 0.6–0.7 ms 20 fps due to additional 7 ms for the center of mass estimation	Some instability with noisy input or close to singular posture	Case by case Trunk and arms
Numeric IK 15 degrees of freedom Four priority levels	Depends on the number of convergence steps (+ latency): 1—1.7 ms, 22 fps 5—6.4 ms, 21 fps 20—26.2 ms, 17 fps	Some instability with noisy input Coherent solution with low-level attraction toward a preferred posture	Generic Can be extended to full body including both legs

Fig. 14 Collision detection induced by the backward movement of the lower body when reaching a target in the forward direction

8 Discussion and conclusion

It can be seen from the examples presented in Sect. 7 that both the analytic and the numeric IK can recover plausible postures from the partial information provided by the vision system. In addition, although we have limited the precision evaluation to the movement performed in Sect. 7.3 (Table 2), the average error norm can be considered as sufficiently small to permit real-time user interaction (for the PIK, we run five convergence steps for all examples). For us, it is more important to provide a globally satisfying posture that can be easily adjusted on the fly, rather than a highly precised one at a bigger computing cost, which would result in a slower frame rate and increased user's stress. Furthermore, the test from Fig. 14 demonstrates that the reconstructed posture is coherent with the natural balance enforcement mechanism where the lower body moves in the opposite direction of the upper body to ensure that the center of mass always projects over the feet supporting area. The resulting feedback is important, as it tells the user about a limitation in a given reach direction due to an obstacle in the opposite direction.

The fact that no information was available to locate the elbow forced each IK approach to make a somewhat arbitrary decision about what was the optimal swivel
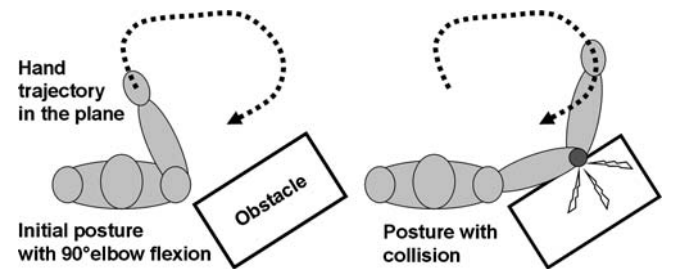
Fig. 15 Collision detection between the elbow and an obstacle located on the side of the user

Fig. 16 Collision-free posture (*left*) and posture causing a collision between the elbow and the obstacle (*right*); the obstacle changes color to indicate the occurrence of the collision
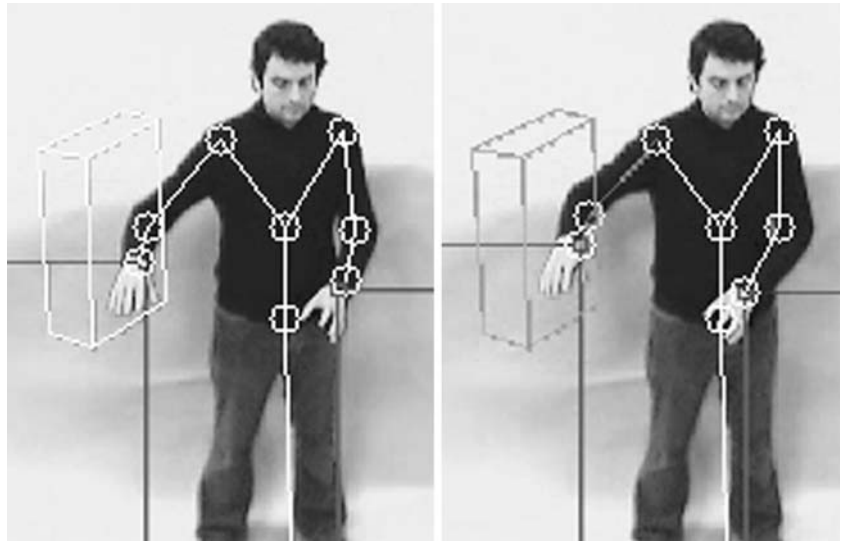
angle for the arm. In the present study, the analytic solution reflects an attraction toward the shoulder mid-range posture while the numeric solution retains an attraction of the upper arm along the body. This can lead to visible difference between the user and the recovered posture. This difference depends on the nature of the movement; at the time being we think there is no shoulder posture that can serve as optimal attractor for all classes of movements. It remains to be evaluated whether the user could be annoyed by this postural difference during real-time interaction. Our guess is, it will depend on the intended application; for example, if the user wishes to pilot the posture of a virtual mannequin moving in a cluttered environment, it is important to offer a finer postural control including the elbow location too. In performance animation contexts where only the hand location is important, both the analytic and the numeric IK solution successfully provide a plausible posture, even when the numeric IK is allowed only very few iterations to converge. In some other contexts where the elbow location conveys some semantic or stylistic clues (e.g., rap singer), it is necessary to integrate additional constraints to recover the user swivel angles too. One solution could be to control the hand orientation so that it matches a target orientation estimated from the vision input. Another possibility to explore would be to constrain elbow effectors to lie inside the body silhouette.

Apart from the plausibility criterion, Table 3 gathers the other criteria used to compare the two IK techniques. Regarding the sensitivity to the noisy vision-based input, the analytic solution is more robust for the torso control as it only exploits the orientation resulting from the center of mass and the head locations, whereas the hand instabilities transfer completely on the arm postures. Conversely, hands position variations have an impact on the whole body posture for the numeric IK approach; they are nevertheless partly smoothed out by the high priority constraint on the center of mass and by the damping factor used to handle the singularities (Maciejewski 1990). The computing cost criterion shows similar performances for both approaches in the present study (on an AMD Athlon 2800 + 2.083 GHz under Windows XP). The performance of the analytic IK is 0.65 ms on average but, compared to the numeric IK, it requires the additional computation of the image center of mass which adds a supplementary cost of 7 ms. On the other hand, one single convergence iteration of the numeric IK costs 1.7 ms, for 15 dofs and 14 constrained dimensions distributed in the three top priority levels (Table 1). It is important to remember that even if a small number of convergence steps is not sufficient for enforcing all constraints, by construction the PIK approach enforces first the highest priority constraints (Baerlocher and Boulic 2004). In the present study, we have chosen to give the highest priority to the center of mass constraint hence favoring the enforcement of the static balance of reconstructed postures; this is important for ensuring the plausibility of the posture. Adding more convergence steps gradually ensures that lower priority constraints are also achieved. As a consequence, the spatial error on the wrist position rapidly decreases (Table 2). Table 3 also reports the frame rates achieved when running one or the other IK approach. All in all, five numeric IK convergence iterations per frame cost roughly the same as one analytic solution update, leading to a refresh rate around 20 fps. This is sufficient for real-time interactions.

We foresee that the following directions have a good potential for improving this technology of full body tracking for real-time interactions:

- First, exploit the body silhouette for inferring a constraint on the elbow location, as this body part has a

direct influence on the plausibility of the resulting posture and on the quality of the wrist position tracking (Fig. 5).

- Second, a better user skeleton is needed both in terms of correspondence with the real skeleton sizes and also in terms of anatomic mobility by including the clavicles, both legs and more vertebrae in the spine.
- Third, in a more general context where the user can move the feet, it will become critical to know which foot is in contact with the floor to adjust the constrain on the center of mass.
- Finally, it can also be desirable to have a parameterized mass distribution model to match a wider population of subjects.

In the longer term, we wish to investigate the problem of controlling the posture of a virtual human that has a different size and body proportions than oneself. This is clearly important for applications evaluating a product for a large population of potential users. Providing an answer to this question will really empower the user through ones full body motion.

To conclude, both the analytic and the numeric IK techniques have the capacity to handle the posture recovery in real-time. The analytic approach strong point is its low computing cost that will scale well for handling more complex skeleton models. The numeric approach strength is its flexibility that ensures a good plausibility of recovered postures by ranking the constraints using strict priority levels.

References

Badler NI, Hollick MJ, Granieri JP (1993) Real-time control of a virtual human using minimal sensors. In: Presence 2(1), MIT Press, Cambridge, pp 82–86. ISSN 1054-7460

Baerlocher P, Boulic R (2004) An inverse kinematic architecture enforcing an arbitrary number of strict priority levels. Visual Comput 20(6):402–417

Bodenheimer R, Rose C, Rosenthal S, Pella J (1997) The process of motion capture: dealing with the data. In: Proceedings of computer animation and simulation. Eurographics Association, pp 3–18

Boulic R, Mas R, Thalmann D (1996) A robust approach for the center of mass position control with inverse kinetics. J Comput Graph 20(5):693–701

Boulic R, Baerlocher P, Rodríguez I, Peinado M, Meziat D (2004) Virtual worker reachable space evaluation with prioritized inverse kinematics. In: Proceedings of the 35th international symposium on robotics, Paris, March 2004

Boulic R, Peinado M, Le Callennec B (2005) Challenges in exploiting prioritized inverse kinematics for motion capture and postural control. Lecture notes in artificial intelligence, Springer, Berlin Heidelberg New York

Bradski GR, Pisarevsky V (2000) Intel's computer vision library. In: Proceedings of the IEEE conference on computer vision and pattern recognition, CVPR00, vol 2, pp 796–797

Chai J, Hodgins JK (2005) Performance animation from low-dimensional control signals. ACM Trans Graph 24(3):686–696

Grochow K, Martin SL, Hertzmann A, Popovic Z (2004) Style-based inverse kinematics. ACM Trans Graph 23(3):522–531

Horain P, Bomb M (2002) 3D model based gesture acquisition using a single camera. In: Proceedings of the sixth IEEE workshop on applications of computer vision, WACV02, pp 158–162

Jojic N, Turk M, Huang TS (1999) Tracking self-occluding articulated objects in dense disparity maps. In: Proceedings of the IEEE international conference on computer vision, ICCV99, vol 1, pp 123–130

Maciejewski AA (1990) Dealing with the ill-conditioned equations of motion for articulated figures. IEEE Comput Graph Appl 10(3):63–71

Moeslund TB, Granum E (2001) A survey of computer vision-based human motion capture. Comput Vis Image Underst 81(3):231–268

Molet T, Boulic R, Rezzonico S, Thalmann D (1999) An architecture for immersive evaluation of complex human tasks. IEEE Trans Rob Autom 15(3):475–485

O'Brien J, BodenHeimer RE, Brostow GJ, Hodgins JK (2000) Automatic joint parameter estimation from magnetic motion capture data. In: Proceedings of Graphics Interface, Morgan Kaufmann Publishers, Mosby, St. Louis, pp 53–60. ISBN 0-9695338-9-6

Sturman D (1998) Computer puppetry. IEEE Comput Graph Appl 18(1):38–45

Tolani D, Goswami A, Badler NI (2000) Real-time inverse kinematics techniques for anthropomorphic limbs. Graph Models 62(5):353–388

Unzueta L, Berselli G, Cazón A, Lozano A, Suescun Á (2005) Genetic algorithms application to the reconstruction of the human motion using a non-invasive motion capture. In: Multibody dynamics, ECCOMAS thematic conference, Madrid, Spain, 21–24 June 2005

Varona J, Buades JM, Perales FJ (2005) Hands and face tracking for VR applications. Comput Graph 29(2):179–187

Wang L, Hu W, Tan T (2003) Recent developments in human motion analysis. Pattern Recognit 36(3):585–601

Wren CR, Clarkson BP, Pentland AP (2000) Understanding purposeful human motion. In: Proceedings of the fourth IEEE international conference on automatic face and gesture recognition. IEEE Computer Society, Grenoble, pp 378–383. ISBN 0-7695-0580-5