Estimating the 3D Shoulder Position using Monocular Vision and a Detailed Shoulder Model

T.B. Moeslund, M. Vittrup, K.S. Pedersen, M.K. Laursen,
M.K.D. Sørensen, H. Uhrenfeldt, and E. Granum
Laboratory of Computer Vision and Media Technology
Aalborg University, 9220 Aalborg East, Denmark

Abstract

Estimating the pose parameters of a human body using monocular computer vision is an interesting but also difficult problem. Introducing assumptions is therefore often necessary to make a certain method work. In this paper we aim at eliminating one of the primary assumptions, namely that of knowing the initial pose parameters. Concretely we present a system to estimate the initial pose of an important anatomic feature - the shoulder. We obtain the 3D position of the shoulder by analysing the projection of two circles described by the hand when moving the outstretched arm in circle arcs. The two projected circles yield two distorted ellipses in the image plane. We correct these ellipses according to a detailed anatomic model of the shoulder. This yields two ellipses in the image plane from where the 3D shoulder position is estimated. The method yields a mean estimation error of $79.7mm$ and a deviation of $51.3mm$ with a distance of $4m$ between the camera and the user.

Keywords: Pose estimation, human motion, the shoulder complex, ellipse fitting.

1 Introduction

A great number of computer vision research papers have addressed the problem of estimating the pose parameters of the human body utilising monocular vision, see [14] for a survey. It appears to be a very hard problem to solve. One of the reasons is the high dimensionality of the solution space when matching the image data to a geometric model of the human body or parts of it. The issue of the high dimensionality is usually addressed by using prediction followed by a local optimisation process. Hence the pose parameters are found as incremental updates with respect to the pose data in the previous image as opposed to absolute pose estimation in each frame. This type of approach has two major drawbacks. Firstly, the initial pose parameters need to be known in advance. Secondly, the approach must never fail to estimate the pose parameters in more than a few consecutive frames, since this will result in a wrong prediction even if good motion models and predictors, e.g. Kalman Filters or Condensation algorithms, are applied. Both drawbacks are often avoided by assuming the initial pose parameters are known [4, 17, 19] and/or that continuous tracking is always possible [2, 7, 18].

In this paper we aim at estimating the initial pose parameters of the shoulder using monocular vision, hence the 3D position of the shoulder. This position is of general interest in HCI applications where the whereabouts of the arms are one of the primary concerns, and especially in gesture recognition where a well-known reference coordinate system (the shoulder) is often required [13].

1.1. The Approach

The user is asked to move his outstretched arm in a circular arc in space. The movement of the hand is captured by a camera and describes an ellipse, or rather a part of an ellipse, in an image sequence. The key idea is now to reverse this process, hence estimating a circle in 3D, or rather its centre, given an ellipse in the image. For this to be applicable several assumptions are introduced. Firstly, the length of the outstretched arm is assumed known. Secondly, the camera needs to be calibrated, and thirdly, the user needs to describe two different circle arcs. Furthermore our approach also assumes a fixed torso position during the circular movements. However, the position of the shoulder itself is not assumed fixed as we correct our measurements according to a detailed anatomic model of the shoulder.

In this paper the design, implementation, and test of the above described approach is presented. The paper is structured as follows. In section 2 the segmentation of the hand is described. In section 3 the ellipse estimation is described together with the correction of the measurements. Section 4 deals with estimating the 3D centre based on the ellipses. In
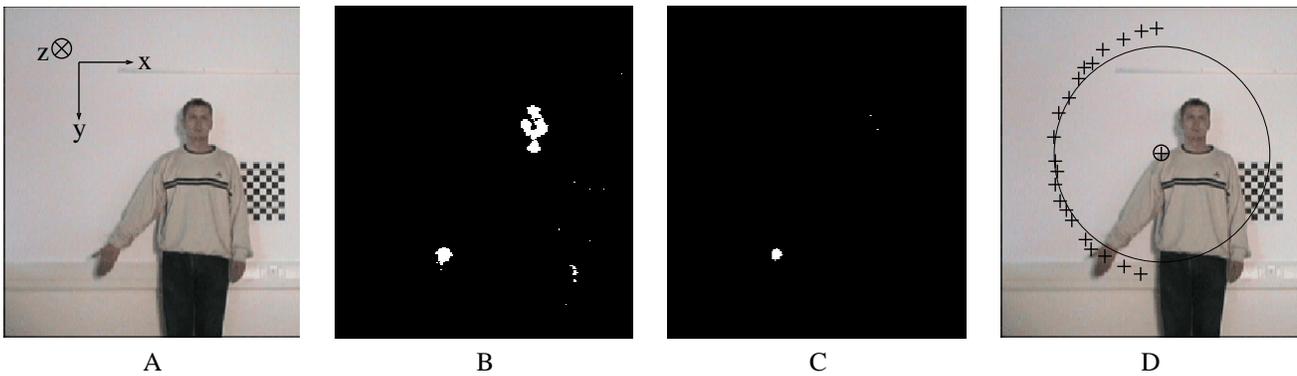
Figure 1. A: A colour input image shown in B/W. B: The input image after applying the colour-based classifier. C: The colour segmented image after motion detection and erosion. D: Input image overlayed by the measurements, the estimated ellipse, and the estimated shoulder position.

section 5 the system is tested, and finally section 6 discusses the approach and the obtained results.

2. Estimating the Positions of the Hand

As the user moves his outstretched arm in a circle in space, the positions of the hand in an image sequence describe the ellipse.

We find the hand using colour information and represent it by its centroid. Hand pixels are manually segmented in a training phase and a classifier is built. The classifier is based on a look-up-table (LUT) which contains information on which colours belong to the hand and which do not. This relatively simple classifier has proven to be more accurate than both a standard box classifier and a Bayes classifier.

In order to suppress the influence of the intensity, different colour representations were tested. The best classifier was found to be based on the YCbCr-representation [8]. An example using this classifier can be seen in figure 1.B.

The classified pixels are processed further by a motion detector, implemented by image-subtraction, hence only the motionless skin pixels are kept. After an erosion, the largest connected object is the hand, see figure 1.C, and represented by its centre of mass. In figure 1.D the centre of mass for the hand in a sequence is superimposed as crosses.

3. Estimating the Ellipse

When image data is used to infer information regarding the 3D world, the camera needs to be calibrated. We use Jean-Yves Bouguet's Matlab implementation [1] of Heikkilä and Silven's method [10]. It defines a world coordinate system, $[x_w \ y_w \ z_w]^T$, with origin in the focal point of a normalised pinhole model. This means that a pixel

$[x_p \ y_p]^T$ is mapped to a line in space as $[x_p \ y_p]^T \rightarrow [x_w/z_w \ y_w/z_w \ z_w]^T$. At $z_w{=}1mm$ a virtual plane is defined having the coordinate system xyz. In the xy plane the pixel representations of the hand-positions form an ellipse - as oppose to a distorted ellipse in the image plane.

To solve the problem of fitting an ellipse to a set of data points we apply the direct fitting approach of [6] using the improvements suggested in [9] to avoid (close to) singularity matrices. This provide us with a fast and more numerically stable solution. The estimated ellipse has the following form on the virtual plane

$$Ax^2 + Bxy + Cy^2 + Dx + Ey + F = 0 \qquad (1)$$

3.1. Correcting the Measurements

The approach described above is based on the assumption that the arm is connected to the shoulder in a ball-and-socket joint, and that the shoulder is fixed with respect to the torso. These assumptions are applied in virtually every single computer vision system utilising a geometric model of a human arm. Besides simplicity, the assumptions are introduced since the coarse motion of the arm is well described in this manner. However, when also fine motion is modelled, and in particular if one desires to estimate the position of the shoulder, the assumptions are false. Since we need rather precise measurements of the ellipse described by the hand we need to introduce a more detailed model of the shoulder joint. The idea is to find the true position of the shoulder joint given the position of the hand and then correct the position of the hand accordingly. To do so we first investigate the anatomy and functionality of the shoulder complex.

The shoulder complex consists of the two bones; the clavicle and the scapula, and the three joints; the sterno-
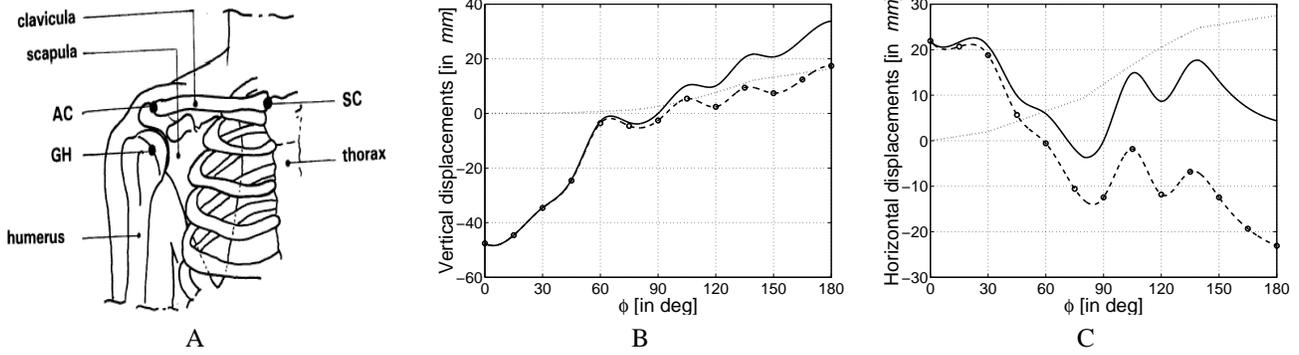
Figure 2. A: The different bones and joints in the shoulder complex. Figure after [3]. B+C: The vertical and horizontal displacements, respectively, as a function of ϕ. The dashed graphs are the displacements of the AC-joint due to the rotation in the SC-joint. The circles are the actual measurements from [5] while the graphs are obtained by spline interpolations. The dotted graphs are the additional displacements of the GH-joint due to rotation in the AC-joint. The solid graphs are the total displacements of the GH-joint with respect to the SC-joint normalised to zero at $\phi = 90°$.

clavicular (SC), the acromio-clavicular (AC), and the gleno-humeral (GH), see figure 2.A. Furthermore, the scapula is connected directly to the thorax via muscles. All three joints functions as ball-and-socket joints each with three degrees-of-freedom (DoF). Together with the DoF provided in the scapular-thorax connection a total of 11 DoF exist [11].

However, since the shoulder complex is a closed kinematic chain these 11 DoF are not independent and in fact the relative pose of the glenoid (the "socket" part of the GH-joint) is well described by only four independent parameters [20]. We model these as rotations around the z- and y-axes, and vertical and horizontal displacements along the y- and x-axes, see figure 1.A for a definition of the axes. The rotations can be governed by the GH-joint by increasing its range accordingly whereas the two displacements can be modelled by two prismatic joints, each having one DoF.

When the upper arm moves the clavicle and scapula follows. In the work by Dvir and Berme [5] the displacements of the AC-joint as a function of the angle between the torso and upper arm, ϕ, is presented, see the dashed lines in figure 2.B and 2.C. To understand how the rotation in the AC-joint adds to the displacements we study the "shoulder rhythm" [20] which describes how the scapula rotates as a function of ϕ. It is usually divided into four phases wherein the relation between the scapula rotation and ϕ is similar. Studying these four phases allow us to calculate the displacements as a function of ϕ. These are shown in figure 2.B and 2.C as the dotted lines. The solid lines in the figures are the sum of the dotted- and dashed lines, hence the displacements of the GH-joint when both the rotation of the clavicle and scapula are taken into account. See [12] for further information.

Since the graphs in figure 2 are given in *mm* we need to map them into pixels. This can be done by knowing the approximate distance from the camera to the user or by measuring a known distance in pixels, e.g. the distance between the hand and the head when the arm is approximately located along the x-axis.

So, given a position of the hand and a mapping into pixels we can estimate ϕ and via the solid graphs in figure 2 find the displacements of the shoulder. These two values are then applied to correct the measured centre of the hand. After this is done for each measurement we obtain the true ellipse. In figure 1.D the measurements are shown together with the ellipse estimated from the corrected measurements.

4. Estimating the 3D Shoulder Position

The circle drawn by the hand in 3D, projects an ellipse on the virtual plane. The ellipse together with a fixed point, denoted the vertex and defined as $[\alpha \ \beta \ \gamma]^T = [0 \ 0 \ -1]^T$, form a general cone embodying the circle. The problem of finding the pose of the circle is equivalent to first finding a plane that intersects the general cone to form a circular graph, and then finding the centre of this graph. The general cone is described as [16]

$$ax^2 + by^2 + cz^2 + 2fyz + 2gzx + 2hxy + $$
$$2ux + 2vy + 2wz + d = 0 \qquad (2)$$

where

$$a = \gamma^2 A \quad b = \gamma^2 C \quad d = \gamma^2 F \quad f = -\gamma E/2 \quad g = -\gamma D/2$$
$$c = F \quad h = \gamma^2 B/2 \quad u = \gamma^2 D/2 \quad v = \gamma^2 E/2 \quad w = -\gamma F$$

3

In equation 2, the second degree terms define the shape of the cone, the terms involving mixed products (e.g. $2fyz$) are related to any rotation it may have and the first degree terms define the translation of the cone vertex. Due to the relatively complex nature of the general cone a number of coordinate transformations are performed to simplify matters.

The first coordinate transformation maps the general cone into the right elliptic cone of the form

$$\lambda_1 X^2 + \lambda_2 Y^2 + \lambda_3 Z^2 = 0 \qquad (3)$$

by first performing a rotation, $[x\ y\ z\ 1]^T = \mathbf{R_1} \cdot [x'\ y'\ z'\ 1]^T$, and then a translation, $[x'\ y'\ z'\ 1]^T = \mathbf{T} \cdot [X\ Y\ Z\ 1]^T$.

The rotation matrix deals with the first six terms of equation 2 and is defined as $\mathbf{R_1} = [\mathbf{i}\ \mathbf{j}\ \mathbf{k}\ \boldsymbol{\sigma}]^T$, where $\mathbf{i} = [i_1\ i_2\ i_3\ 0]^T$, $\mathbf{j} = [j_1\ j_2\ j_3\ 0]^T$, $\mathbf{k} = [k_1\ k_2\ k_3\ 0]^T$, and $\boldsymbol{\sigma} = [0\ 0\ 0\ 1]^T$. To estimate $\mathbf{R_1}$ we write the first six terms in equation 2 as $\mathbf{q}^T \mathbf{P} \mathbf{q}$ where $\mathbf{q} = [x\ y\ z]^T$, $\mathbf{P} = [\mathbf{p_1}\ \mathbf{p_2}\ \mathbf{p_3}]$, $\mathbf{p_1} = [a\ h\ g]^T$, $\mathbf{p_2} = [h\ b\ f]^T$, and $\mathbf{p_3} = [g\ f\ c]^T$. We can now find the column vectors of $\mathbf{R_1}$ as the eigenvectors of $\mathbf{P}$, and the coefficients in equation 3 as the corresponding eigenvalues [15].

Applying $\mathbf{R_1}$ to equation 2 yields

$$\lambda_1 x'^2 + \lambda_2 y'^2 + \lambda_3 z'^2 + 2(ui_1 + vj_1 + wk_1)x' + 2(ui_2 + vj_2 + wk_2)y' + 2(ui_3 + vj_3 + wk_3)z' + d = 0 \qquad (4)$$

where the cone has its principal axis parallel to the z'-axis.

Next the cone is translated such that its vertex lies in the origin according to the XYZ coordinate system thus finishing the reduction of equation 2 to equation 3. This translation, $\mathbf{T}$, can be found by comparing equations 3 and 4 [16].

Since equation 3 is a right elliptic cone, two of the three λ_i values will be positive and one negative. We assign positive values to λ_1 and λ_2 yielding a right elliptic cone having the Z axis as its principal axis.

4.1. Estimating the Intersection Plane

The next step is to find a plane such that, when intersecting with the right elliptical cone, a circular section appears.

In particular, the aim is to find the coefficients of the intersection plane, equation 5, intersecting the cone to generate the equation of a circle.

$$lX + mY + nZ = p \qquad (5)$$

where $[l\ m\ n]^T$ has length equal to 1.

Using yet another rotation matrix the new Z axis, Z', becomes normal to the plane defined in equation 5. Thus the plane in the new coordinate system will be represented by the equation $Z'=p$. The rotation is defined as

$[X\ Y\ Z\ 1]^T = \mathbf{R_2} \cdot [X'\ Y'\ Z'\ 1]^T$ where $\mathbf{R_2} = [\mathbf{a}\ \mathbf{b}\ \mathbf{c}\ \boldsymbol{\sigma}]$, $\mathbf{a} = [a_1\ a_2\ a_3\ 0]^T$, $\mathbf{b} = [b_1\ b_2\ b_3\ 0]^T$, $\mathbf{c} = [c_1\ c_2\ c_3\ 0]^T$, and $\boldsymbol{\sigma} = [0\ 0\ 0\ 1]^T$. $\mathbf{c} = [l\ m\ n\ 0]^T$ and $\mathbf{a}$ and $\mathbf{b}$ can be defined based on $\mathbf{c}$ and the orthogonality constraint.

Finding the coefficients of equation 5 is done via a method similar to the one used to find the centre of the circle, which is described next. The description is therefore left out here and the reader is referred to 4.2 and [16] for details. It should, however, be stated that the method results in two plausible solutions, i.e. $\mathbf{R_{2a}}$ and $\mathbf{R_{2b}}$.

4.2. Estimating the Circle Centre

Having found a plane which gives a circle when intersecting with the cone, finding the position of the circle centre remains. Moving the plane along the Z' axis results in different sizes of circles, as the cone obviously expands as the distance to the vertex grows. Knowing the radius of the circle, R, provides information to calculate where on the Z' axis the intersection plane should be located.

Applying $\mathbf{R_2}$ to equation 3 yields in general a quadratic equation of the same form as equation 1. We know that the actual shape is a circle, hence $A = C$ and $B = 0$. The (circle) equation obtained by applying $\mathbf{R_2}$ to equation 3 can therefore be expressed as [16]

$$\left(X' + \frac{pD'}{A'}\right)^2 + \left(Y' + \frac{pE'}{A'}\right)^2 = \frac{p^2 D'^2}{A'^2} + \frac{p^2 E'^2}{A'^2} - \frac{p^2 F'}{A'} \quad (6)$$

where

$$
\begin{aligned}
A' &= \lambda_1 a_1^2 + \lambda_2 a_2^2 + \lambda_3 a_3^2 \\
D' &= \lambda_1 a_1 c_1 + \lambda_2 a_2 c_2 + \lambda_3 a_3 c_3 \\
E' &= \lambda_1 b_1 c_1 + \lambda_2 b_2 c_2 + \lambda_3 b_3 c_3 \\
F' &= \lambda_1 c_1^2 + \lambda_2 c_2^2 + \lambda_3 c_3^2
\end{aligned}
$$

The coordinates of the centre, $[X_0'\ Y_0'\ Z_0']^T$, are found from equation 6 together with the fact that the radius of the circle is given as R and that $Z' = p$, hence

$$
X_0' = -\frac{Z_0' D'}{A'} \qquad Y_0' = -\frac{Z_0' E'}{A'}
$$

$$
Z_0' = \pm \frac{A' R}{\sqrt{D'^2 + E'^2 - A'F'}}
$$

As can be seen Z_0' has two possible values of which only one is valid. As the circle is performed in front of the camera, Z_0' is chosen such that when mapped back to the xyz coordinate system by $[x\ y\ z]^T = \mathbf{R_1}\mathbf{T}\mathbf{R_2} \cdot [X_0'\ Y_0'\ Z_0']^T$, z will be positive, i.e. in front of the camera. The above must be done for each of the two solutions found in section 4.1.
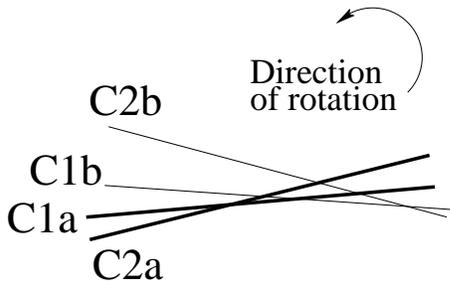
Figure 3. The four plausible circles seen from above. The boldfaced lines are the correct circles.

4.3. Finding the Correct Solution

To determine which solution is correct the user is asked to perform two different circles where the second must be closer to the camera than the first. This yields four possible scenarios where one is shown in figure 3.

The figure illustrates the four circles produced by the method seen from above. For simplicity the circles are only panned with respect to each other, hence they can be represented by lines. The boldfaced lines indicate the correct circles, where $C1a$ is the first circle performed and $C2a$ is the second. Since we know the rotation direction between the two circles we can conclude that $C2a$ contains the correct centre since $C2b$ is unreachable by rotating $C1a$ or $C1b$ in the illustrated direction.

Representing the scenario in figure 3, with respect to the rotation direction, as $\Pi = \{C2b, C1b, C1a, C2a\}$ the following rule can be defined

$$\text{correct circle} = \begin{cases} C2a & \text{if } \Pi = \{C2b, C1b, C1a, C2a\} \text{ or} \\ & \quad \Pi = \{C2b, C1a, C1b, C2a\}, \\ C1a & \text{if } \Pi = \{C1a, C2a, C2b, C1b\} \text{ or} \\ & \quad \Pi = \{C1a, C2b, C2a, C1b\}. \end{cases}$$

After the correct centre has been found in xyz coordinates, it is mapped to world coordinates $x_w y_w z_w$ by translating the z coordinate $1mm$, see section 3.

5. Tests and Results

5.1. Testing the Ellipse Fitting Algorithm

Synthetic ellipses are used to evaluate the ellipse fitting method described in section 3. It was observed that the method tends to estimate too large ellipses when few samples are used. In the general case when sufficient samples

are applied the method has a low eccentricity bias [6]. Combining this with only applying samples from the left half of the ellipse results in smaller and smaller ellipses (to a certain degree) as the number of samples increase. Since the samples describe the entire range of the ellipse in the y-direction, see figure 1, the resulting ellipses, besides getting smaller and smaller, have their centres shifted to the left.

5.2. Testing the 3D Shoulder Position

The 3D pose estimation method described in section 4 is tested in the following manner. A synthetic circle is projected onto the virtual plane yielding an ellipse. A number of points, η, from the left arc of the ellipse are sampled and infected with noise by first randomly selecting a direction in the interval $\{0, 2\pi\}$ using a uniform distribution, and secondly finding the magnitude by sampling a Gaussian distribution which parameters are found by mapping the mean and deviation errors of the centroid (found during training to: mean = 2.4 and deviation = 1.97 pixels) onto the virtual plane. These points are fed to the estimation algorithm which produce two shoulder candidates. The same is carried out for a different synthetic circle and a single shoulder candidate can be estimated, see section 4.3.

The parameters of the synthetic circles are chosen randomly within a range similar to that of the application. For each value of η within the interval $\{6, 25\}$ 1000 estimates of the shoulder position are found. The error is defined as the ground truth subtracted from the estimated position. In figure 4 the statistics of this test are shown.

In figure 4.D-F it can be seen that the deviation of the errors decreases as the number of samples increase. After the 25 samples the deviation is below $20mm$ for all parameters. Figure 4.A is a result of the incorrect estimate of the centre of the ellipse described above. As more samples are used the centre is drawn to the left. Figure 4.C can also be explained by the incorrectness of the ellipse fitting. First the size of the estimated ellipse is too large resulting in a shoulder estimate too close to the camera. As more samples are used the size of the ellipse is under-estimated resulting in a shoulder estimate too far away from the camera.

5.2.1 Testing the 3D Position using a Cardboard

The next test is conducted on circle data drawn on a cardboard. The circle has a radius similar to the length of a human arm and is placed $4m$ from the camera. The cardboard is mounted on a large camera tripod which can be tilted, rolled, and panned. A calibration pattern, see figure 1.A, is placed on the cardboard having its upper left corner in the centre of the circle. Using the toolbox [1] the centre of the circle can be estimated. 35 different poses of the circle are generated. A number of samples are hand-segmented from
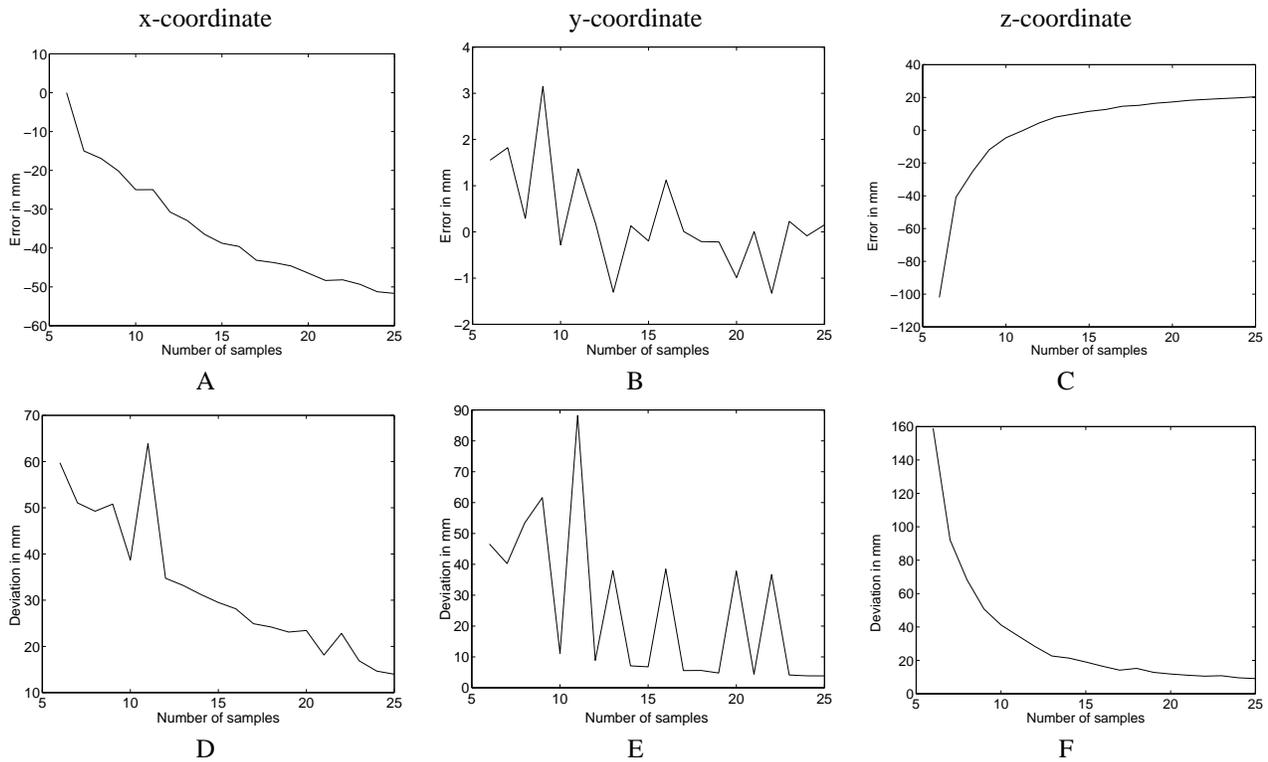
Figure 4. A-C: The mean errors of the estimated x, y, and z coordinates, respectively. D-F: The deviation of the estimated x, y, and z coordinates, respectively.

the image and infected with noise as described in section 5.2, and processed by the system. The magnitude of the error vector between the estimated centre and the one found from the toolbox are recorded and the statistics are calculated to a mean of $42.2mm$ and a deviation of $18.9mm$.

5.3. Testing on Real Data

The final test is conducted on real persons performing circle-movements as shown in figure 1. The ground truth is defined as the centre of the arm where it is attached to the torso, when the arm is held in a horizontal position and parallel to the camera. The ground truth x- and y-coordinates are found using the toolbox. The z-coordinates are found by measuring the perpendicular distance from the shoulder to the back wall. Images of a user performing the two circle-movements are captured and processed. The magnitude of the error vector is recorded for a number of test persons in different poses and the statistics are a mean of $79.7mm$ and a deviation of $51.3mm$.

6. Discussion and Conclusion

At first glance figure 4 might indicate an unstable system since on one hand figures 4.D-F suggest using as many samples as possible to obtain a small deviation whereas figures 4.A-C suggest using different number of samples for the estimation of the different parameters. However, since the graphs as such are stable, i.e. invariant to different test data, the following is done. We apply as many samples as possible to obtain a stable result and then correct this result according to a LUT designed from figure 4.A-C. For example, if we have 20 samples[1] we first use them to estimate $[x \; y \; z]^T$ and thereafter correcting the final shoulder position using the LUT, hence $[x+46 \; y+0 \; z-18]^T$. Altogether we obtain close to zero mean estimates with a deviation depending on the number of samples, see figure 4.D-F[2].

An even more stable solution may be obtained if the LUT is replaced with an iterative method to estimate the ellipse, perhaps with the result of the current method as an initial guess. Perhaps the method could also utilise information re-

---

[1]The number of present samples depends on the speed of the movement of the arm, the image processing time, and the framerate of the system.

[2]The LUT was applied to obtain the results which were presented in section 5.2.1 and section 5.3.

6

garding the position of the head since this is relatively fixed with respect to the shoulder.

Tests on the cardboard yield an increase in the error rate compared to the ideal data. This is primarily due to the non-ideal data, but also due to the insufficient pinhole model. In the final test on users the error rate is increased further due to incorrect ground truth and violations of the assumptions. The violations occur if the arm is not held completely outstretched during the movement or if the person translate the torso during the movement.

Nearly all the tests yield a relatively symmetric distribution having a significant peak around the mean and then a few relatively large errors. Many of the relatively large errors can be avoided by only accepting a shoulder position if its two components (the solutions from the two circles) are similar. In practice the system could keep asking the user to perform circles until two with close centres are found.

Altogether it is concluded that the system is capable of estimating the 3D position of the shoulder using just one camera. This is done with a relative good accuracy, compared to the distance of $4m$ between the camera and the user. The detailed shoulder model applied to correct the measurements is perhaps the single-most important factor for the success of our system. This novel modelling scheme can hopefully provide other computer vision systems with a more detailed model of the human arm/shoulder without increasing the number of independent parameters.

References

[1] J.-Y. Bouguet. Camera Calibration Toolbox for Matlab. *www.vision.caltech.edu/bouguetj/calib_doc/index.html*.

[2] C. Bregler and J. Malik. Tracking People with Twists and Exponential Maps. In *CVPR*, California, 1998.

[3] K. Breteler, C.W. Spoor, and F.C.T. Van der Helm. Measuring Muscle and Joint Geometry Parameters of a Shoulder for Modelling Purposes. *Journal of Biomechanics*, 32(11), 1999.

[4] Q. Delamarre and O. Faugeras. 3D Articulated Models and Multi-view Tracking with Silhouettes. In *ICCV*, Corfu, Greece, September 1999.

[5] Z. Dvir and N. Berme. The Shoulder Complex in Elevation of the Arm: A Mechanism Approach. *Journal of Biomechanics*, 11, 1978.

[6] A. Fitzgibbon, M. Pilu, and R.B. Fisher. Direct Least Squares Fitting of Ellipses. *PAMI*, 21(5), 1999.

[7] D.M. Gavrila and L.S. Davis. 3-D Model-Based Tracking of Humans in Action: A Multi-View Approach. In *CVPR*, San Francisco, 1996.

[8] R.C. Gonzalez and R.E. Woods. *Digital Image Processing*. Addison-Wesley Publishing Company, 1993.

[9] R. Halíř and J. Flusser. Numerically Stable Direct Least Squares Fitting of Ellipses. In *The 6th Int. Conf. in Central Europe on Computer Graphics and Visualization.*, Plzeň, Czech Republic, 1998.

[10] J. Heikkilä and O. Silven. A Four-step Camera Calibration Procedure with Implicit Image Correction. In *CVPR*, Puerto Rico, 1997.

[11] W. Maurel. *3D Modeling of the Human Upper Limb including the Biomechanics of Joints, Muscles and Soft Tissues*. PhD thesis, Laboratoire d'Infographie - Ecole Polytechnique Federale de Lausanne, 1998.

[12] T.B. Moeslund. Modelling the Human Arm. Technical Report CVMT 02-01, ISSN 0906-6233, Laboratory of Computer Vision and Media Technology, Aalborg University, Denmark, 2002.

[13] T.B. Moeslund and E. Granum. 3D Human Pose Estimation using 2D-Data and an Alternative Phase Space Representation. In *CVPR Workshop on Human Modeling, Analysis and Synthesis*, South Carolina, 2000.

[14] T.B. Moeslund and E. Granum. A Survey of Computer Vision-Based Human Motion Capture. *Computer Vision and Image Understanding*, 81(3), 2001.

[15] B. Nobel and J.W. Daniel. *Applied Linear Algebra*. Englewood Cliffs, NJ: Prentice-Hall, 1977.

[16] R. Safaee-Rad, I. Tchoukanov, K.C. Smith, and B. Benhabib. Three-Dimensional Location Estimation of Circular Features for Machine Vision. *Transactions on Robotics and Automation*, 8(5), 1992.

[17] H. Segawa, H. Shioya, N. Hiraki, and T. Totsuka. Constraint-Conscious Smoothing Framework for the Recovery of 3D Articulated Motion from Image Sequences. In *Automatic Face and Gesture Recognition*, Grenoble, France, 2000.

[18] H. Sindenbladh, F. De la Torre, and M.J. Black. A Framework for Modeling the Appearance of 3D Articulated Figures. In *Automatic Face and Gesture Recognition*, Grenoble, France, 2000.

[19] S. Wachter and H.-H. Nagel. Tracking Persons in Monocular Image Sequences. *Computer Vision and Image Understanding*, 74(3), 1999.

[20] V.M. Zatsiorsky. *Kinematics of Human Motion*. Champaign, IL: Human Kinetics, 1998.