Towards markerless visual servoing of grasping tasks for humanoid robots

Pedro Vicente[1], Lorenzo Jamone[2] and Alexandre Bernardino[1]

***Abstract*— Vision-based grasping for humanoid robots is a challenging problem due to a multitude of factors. First, humanoid robots use an "eye-to-hand" kinematics configuration that, on the contrary to the more common "eye-in-hand" configuration, demands a precise estimate of the position of the robot's hand. Second, humanoid robots have a long kinematic chain from the eyes to the hands, prone to accumulate the calibration errors of the kinematics model, which offsets the measured hand-to-object relative pose from the real one. In this paper, we propose a method able to solve these two issues jointly. A robust pose estimation of the robot's hand is achieved via a 3D model-based stereo-vision algorithm, using an edge-based distance transform metric and synthetically generated images of a robot's arm-hand internal computer-graphics model (kinematics and appearance). Then, a particle-based optimisation method adapts on-line the robot's internal model to match the real and the synthetically generated images, effectively compensating the kinematics calibration errors. We evaluate the proposed approach using a position-based visual-servoing method on the iCub robot, showing the importance of the continuous visual feedback in humanoid grasping tasks.**

I. Introduction

Humanoid robots are raising great interest in the research community. They are versatile platforms and can be used in diverse application scenarios since the match with human dimensions and degrees-of-freedom facilitates the operation in human-made environments and with human-made objects and tools. However, these robots have complex mechanical structures and long kinematic chains (e.g. the iCub humanoid robot [1] has 53 mechanical degrees-of-freedom – see Fig. 1) which are difficult to model and calibrate in order to perform even the most basic tasks with great accuracy. Precision reaching and grasping are examples of challenging tasks due to hard-to-model characteristics of the humanoid kinematic chains (e.g elasticity), changes that occur due to environmental conditions (e.g. material dilation due to temperature), or conditions related to the operation of the system, both in the long term (e.g. joints drift and misalignment due to wear and mechanical stress) and in the short term (e.g. bending due to payload and gravity).

Indeed, the new emerging market targeting human-robot interaction and robots with a large number of degrees-of-freedom requires real-time on-line strategies to continuously adapt to changes in the environment. Anyway, there are always unknown factors that will be left out of the internal model and will result in residual uncertainty on the hand-to-object relative pose. If this uncertainty is higher that the required precision in the manipulation, again failure is imminent. Such residual uncertainties must be solved locally using sensor feedback, for instance in a visual-servoing framework [2]. Visual-servoing methods constantly measure the relative pose between the robot's end-effector and the object of interest and control the robot's arm to reduce this error in a form that is robust to small calibration errors.

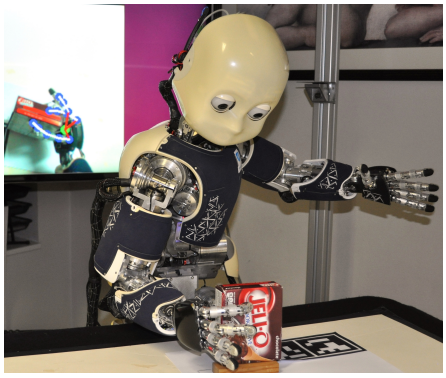


Fig. 1: The iCub humanoid robot performing a precise grasping task by markerless visual servoing.

Surprisingly, very few humanoid robotics works employ continuous visual feedback on the reaching and grasping tasks [3]. Most works rely on an open-loop approach, where the robot looks once to the scene, computes the relative pose between the hand and the object, then drives the arm to the object position without using visual feedback along the process (refer to Sec. II for more details). This is probably due to the difficulty in obtaining a reliable estimate of the robot's hand pose from visual measurements. In fact, the most common applications of visual servo control exploit the "eye-in-hand configuration", where the camera is rigidly attached to the end-effector, so visual perception of the end-effector pose is not required. However, humanoid robots have cameras in the head ("eye-to-hand" configuration), thus the computation of the relative pose between end-effector and objects requires visual perception of the hand. Although some works try to address this problem using special markers in the end-effector to facilitate pose measurements, this approach is not practical in small multi-fingered hands due to the difficulty in affixing the markers. Furthermore, small markers still carry significant uncertainties in rotation.

[1] P. Vicente and A. Bernardino are with the Institute for Systems and Robotics, Instituto Superior Técnico, Universidade de Lisboa, Lisbon, Portugal. {pvicente,alex}@isr.tecnico.ulisboa.pt

[2] L. Jamone is with ARQ (Advanced Robotics at Queen Mary), School of Electronic Engineering and Computer Science, Queen Mary University of London, UK and with Institute for Systems and Robotics, Instituto Superior Técnico, Universidade de Lisboa, Lisbon, Portugal. l.jamone@qmul.ac.uk

In this work, we propose a simultaneous visual-servoing and markerless on-line calibration system for reaching and grasping tasks on the iCub robot. The system builds on two key elements developed in our previous work: (i) a real-time 3D markerless model-based stereo-vision pose estimation of the robot's hand [4], and (ii) a robust particle based-optimisation method that continuously estimates the robot's calibration errors modelled as joints offsets [5]. In the current work, we have developed a position-based "eye-to-hand" visual servoing method based on the previous components and show, for the first time, an automated grasping system with continuous visual feedback operating in the iCub robot.

II. Related work

Real-time reaching/grasping tasks in robots are performed, normally, without any feedback control approach [6], [7], [8], [9], [10], mainly to speed-up the reaching process. According to [3], very few methods of grasping take advantage of vision to correct robotic hand poses. In [6] open-loop grasping of kitchenware objects is performed; however, some experiments failed due to undesired "contact between the hand and the object", which could be mitigated with a visual-control approach. Kim et al. [7] propose to catch objects in flight exploiting the inverse kinematics of a robotic arm, achieving satisfactory results. Although the iCub robot is used for the simulations, the real world experiments are performed with a Kuka industrial arm, whose analytical model is very reliable; arguably, it would have been very challenging to reproduce similar results with the iCub robot, whose kinematic chain is more difficult to model accurately. Also, Leidner et al [9], [10] resort to a precise robotic platform to perform feed-forward control; however, the authors state that their work should be extended to integrate feedback-loop based on visual perception.

Visual servoing is a feedback closed-loop control strategy based on visual data [2]. Most of the visual servoing methodologies are based on eye-in-hand control, where the cameras are attached to the robot hand and local visual features extracted from the object are used to drive the arm motion (e.g the works [11], [12], [13]). One limitation of this strategy is that only a partial view of the scene is available (i.e. the part in front of the hand), and therefore the trajectory of the robot arm should be limited to keep the target object visible; also, if the camera gets very close to the object some global visual information about the object might be lost (e.g. object shape, contours). Another approach, which is typical in humanoid robots, is to mount the cameras in the eyes/head - the eye-to-hand configuration. This configuration is more biologically inspired, it offers a global sight of the task space and it allows to selectively direct the robot attention (independently from the arm motion). However, it requires estimating the pose of both the robot hand and the target object from vision, which are challenging problems. Moreover, the long camera-to-hand kinematic chain, and the fact that the eyes and head can move, introduce serious calibration issues, making it very hard to maintain the reference frames of the cameras and of the hand well aligned. A typical solution to alleviate this problem is to use markers in the robot hand in order to estimate its pose [14], [15], [16], [17], [18], [19], [20], [21]. In [14] the use of a single camera and a landmark (a light bulb emitting red) in the hand, together with the reflection of a flat mirror, improved the 3D estimation of the hand position in the world. In [15], [16], [17], [18], [19], [20], artificial markers are attached to the robot wrist (a check pattern in [15] and a coloured ball in [16], [17], [18], [19], [20], allowing to execute accurate reaching/grasping [15], [16], [17], [18], also with whole-body movements [19] and by using tools [20]. The humanoid robot REEM was used in [21] to perform reaching and grasping tasks with visual feedback, using markers on the hand and on the objects.

Markerless robotic arm posture estimation was been proposed in some recent works using 3D-vision sensors [22], [23]. In [22] the 3D robot model is compared with the point-cloud of the real arm and properly adapted. In [23] the point-cloud is used to train two random forests; one to differentiate between background and arms, and another to estimate the arm posture. Gratal et al [24] use 2D features and an optimisation strategy based on gradient descent to realise virtual visual servoing [25]; this work was extended to include depth information in [26].

Our approach is also based on 2D visual features (either silhouette segmentation or edge extraction); however, we perform optimisation based on particle filtering, which differently from gradient descent is not prone to converge to local minima. In previous work, we provide details on how the GPGPU implementation of our system can achieve real-time hand pose estimation and kinematic chain calibration simultaneously and effectively in the iCub robot [4], [5]. In this paper we show how this system can support markerless visual servoing, allowing to perform precise reaching and grasping in the real world.

III. Proposed Methodology

Let us consider the problem of robotic reaching and grasping. Reaching consists in moving the hand from an initial configuration $\mathbf{T}_h$ to a desired configuration $\mathbf{T}_d$ where the hand is positioned in an adequate pose to grasp the object by closing the fingers under some control law. Configurations $\mathbf{T}_h$ and $\mathbf{T}_d$ are elements of $SE(3)$, the special Euclidean group, and represent the coordinate transformation from a source reference frame (here the initial and desired hand configurations) to a common reference frame, in our case the left eye of the robot. A common representation for elements of $SE(3)$ is that of 4×4 matrices of the form:

$$\mathbf{T} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix} \tag{1}$$

where $\mathbf{R}$ is a 3×3 rotation matrix, $\mathbf{t}$ is a 3×1 translation vector, and $\mathbf{0}$ is a 1×3 vector of zeros. [1].

[1]For simplicity, and because collision avoidance is not in the scope of this work, we assume that the space between the initial and final configurations of the hand is free of (self-) collisions in a production system these conditions must be checked carefully and, if not met, use a planning algorithm (e.g. the RRT algorithm [27]) to drive the robot to a configuration where these assumptions are met.

To drive the hand from initial to desired configuration one has to cancel the error transformation: ${}^{d}\mathbf{T}_h = (\mathbf{T}_d)^{-1}\,\mathbf{T}_h$.

However, in practice is difficult to estimate ${}^{d}\mathbf{T}_h$. Here we analyse the problems arising from errors in the estimate of $\mathbf{T}_h$. Methods to estimate $\mathbf{T}_d$, which basically depend on the estimation of the object pose, are out of the scope of this work (several methods exist for pose estimation of rigid objects from CAD models [28]).

A common way to estimate $\mathbf{T}_h$ is to use the robot's internal model kinematics function. Let $\mathcal{K}(\mathbf{q})$ be the kinematics function that transforms points from the robot's hand to the left eye, where $\mathbf{q}$ is the vector of robot's joint angles. An estimate of the hand pose can be obtained by: $\mathbf{T}_h^{[\text{kin}]} = \mathcal{K}(\mathbf{q})$. However, several sources of calibration errors may exist in the kinematics transformation, from errors in its parameters to non-modeled aspects. Particularly in long kinematic chains (as our case) these errors may significantly affect the estimate of the hand pose $\mathbf{T}_h^{[\text{kin}]}$.

A. Internal Model Calibration

The visual feedback is used to constantly update the robot's internal model. In our previous papers, we provide details of this adaptation process [5] and its generalisation capabilities [4], that are summarised hereinafter.

We encode the calibration errors in a set of parameters $\boldsymbol{\beta}$ representing offsets in the robot's arm joints, i.e.: $\mathbf{q}^r = \mathbf{q} + \boldsymbol{\beta}$, where $\mathbf{q}^r$ are the real angles and $\mathbf{q}$ are the measured angles. Given an estimate of the joint offsets $\hat{\boldsymbol{\beta}}$, a better end-effector's pose estimate can be computed by:

$$\mathbf{T}_h^{[\text{cal}]} = \mathcal{K}(\mathbf{q} + \hat{\boldsymbol{\beta}}) \tag{2}$$

To estimate the parameters $\boldsymbol{\beta}$ we compare the current images acquired by the cameras with images synthetically generated by a graphics game engine (Unity®) and the CAD model of the robot. The search for the set of values $\boldsymbol{\beta}$ that provide the best match between the real and the synthetic images is performed with a Sequential Monte Carlo method similar to a particle filter adapted to the estimate of constant state vectors (parameter estimation) [29].

Let us consider distribution $p(\boldsymbol{\beta}_t|\mathbf{q}_{1:t}, \mathbf{y}_{1:t})$ that represents our belief on the values of $\boldsymbol{\beta}$ at time t given all past observations of the joint encoder angles $\mathbf{q}_{1:t}$, and acquired images $\mathbf{y}_{1:t}$. This distribution is approximated by a set of M samples (particles): $\mathbf{B}_t := \left\{\boldsymbol{\beta}_t^{[1]}, \boldsymbol{\beta}_t^{[2]}, \boldsymbol{\beta}_t^{[3]}, \cdots, \boldsymbol{\beta}_t^{[M]}\right\}$, with an associated importance weight $\omega^{[m]}$:

$$\omega^{[m]} = p(\mathbf{y}_t|\mathbf{q}_t, \boldsymbol{\beta}_t^{[m]})\ ,\ m = 1, \cdots, M \tag{3}$$

The likelihood function (3) is described in Sec.III-B, Eq. (4). The set of particles is then re-sampled according to the importance weights to replicate samples with high likelihood and remove samples with low likelihood. Finally, an artificial dynamics is introduced in the transition model of the parameters $\boldsymbol{\beta}$ defined as: $\boldsymbol{\beta}_t = \boldsymbol{\beta}_{t-1} + \xi$, where ξ is an artificial dynamic noise which decreases when t increases. The particles are thus modified using: $\boldsymbol{\beta}_t^{[m]} \leftarrow \boldsymbol{\beta}_t^{[m]} + \xi$. The cycle composed of the steps for (1) importance weight computations, (2) re-sampling and (3) artificial noise injection, can be repeated a few times to improve convergence.

We need to compute our best guess $\hat{\boldsymbol{\beta}}$ from the particle distribution to calculate the end-effector's pose estimate (Eq. (2)). Instead of choosing the particle with the highest weight in each time step, we compute a kernel density estimation to smooth the particles' weight according to the information of neighbour particles. The best guess will be the particle with the highest smoothed weight ($\omega'^{[i]}$): $\omega'^{[i]} = \omega^{[i]} + \alpha \cdot \frac{1}{M} \sum_{m=0}^{M} \omega^{[m]} \cdot K(\boldsymbol{\beta}^{[i]}, \boldsymbol{\beta}^{[m]})$, where $\omega^{[i]}$ is the particle likelihood, α is a smoothing parameter, M is the number of particles and $\boldsymbol{\beta}^{[i]}$ is the particle we are smoothing. The sum term is the influence of the neighbors in the score of particle i. K is a Gaussian Kernel specifying the influence of one particle in others.

B. Observation model

We exploit the edge information extracted from images proposed in [5]. The average distance between the edges of the real image to the closest edge of the virtual image is denoted by $\bar{d}$. The Distance Transform metric proposed by Borgefors ([30]) is used to compute $\bar{d}$. The Distance Transform (DT) consists in the application of an edge detector to the image (e.g. [31]) and then, for each pixel, compute its distance to the closest edge point. This distance has a minimum of 0 pixel and a maximum of 255 pixel since the DT result is an 8-bit single-channel image. Let $\mathbf{D}(\mathbf{y})$ be the Distance Transform of the real images and $\mathbf{E}(\hat{\mathbf{y}}^{[m]})$ be the edge map (binary image indicating the edge pixels) of the virtual image generates with robot configuration $\mathbf{q} + \boldsymbol{\beta}^{[m]}$.

The average distance, $\bar{d}^{[m]}$ for each particle, can be efficiently computed using the Chamfer matching distance ([32]): $\bar{d}^{[m]} = \frac{1}{k} \cdot \sum_{i=0}^{N} \mathbf{E}\left(\hat{\mathbf{y}}^{[m]}(i)\right) \cdot \mathbf{D}\left(\mathbf{y}(i)\right)$, where k is the number of edge pixels in the virtual image, i is an index that runs over all pixels, and N is the total number of pixels.

A perfect match between the real and virtual images will correspond to $\bar{d} = 0$ whereas bad matches will correspond to large values of $\bar{d}$. The likelihood function become:

$$p(\mathbf{y}_t|\mathbf{q}_t, \boldsymbol{\beta}_t^{[m]}) \propto \exp^{-\lambda_{\text{edge}} \cdot\ \bar{d}} \tag{4}$$

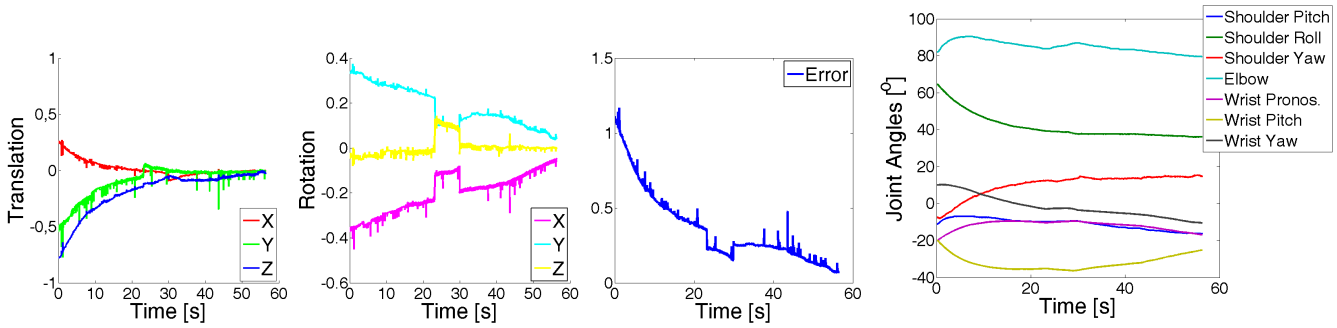
where λ_{edge} is a tuning parameter to control sensitivity in the distance metric.

C. Visual Servoing

Visual Servoing, also known as visual servo control, is a feedback closed-loop control strategy based on visual data. In this work, we use an eye-to-hand configuration, where the cameras looking at the scene are not attached to the end-effector and can observe it. In our case, we have a humanoid robot and the cameras are in the eyeballs.

In this work, we follow the usual control law in eye-to-hand [2] position-based visual servoing [33].

Following the notation from the previous section, the error to be minimised is the transformation from current to the



(a) Translation errors (b) Orientation errors (c) Norm of the error (d) Joint angles

Fig. 2: Example of visual servoing convergence during one experiment.

desired hand pose ${}^{d}\mathbf{T}_{h}$ and can be written as:

$$\mathbf{e} = [\mathbf{t}, \theta\mathbf{u}] \tag{5}$$

where $\mathbf{t}$ is the translation component and $\theta\mathbf{u}$ the rotation in axis-angle notation. The relationship between the time derivation ($\dot{\mathbf{e}}$) and the joint velocities ($\dot{\mathbf{q}}$) of the robot arm can be expressed as:

$$\dot{\mathbf{e}} = \mathbf{J}_s(\mathbf{q})\dot{\mathbf{q}} \tag{6}$$

where $\mathbf{J}_s(\mathbf{q})$ is the feature Jacobian matrix defined as:

$$\mathbf{J}_s(\mathbf{q}) = \mathbf{Le} \cdot {}^{d}\mathbf{V}_l \cdot {}^{l}\mathbf{J}_h(\mathbf{q}) \tag{7}$$

where the subscript l denotes the left eye reference frame, ${}^{l}\mathbf{J}_h(\mathbf{q})$ is the robot Jacobian evaluated at the current configuration $\mathbf{q}$, and ${}^{d}\mathbf{V}_l$ is the spatial motion transform matrix:

$${}^{d}\mathbf{V}_l = \begin{bmatrix} {}^{d}\mathbf{R}_l & [{}^{d}\mathbf{t}_l]_\times \cdot {}^{d}\mathbf{R}_l \\ \mathbf{0} & {}^{d}\mathbf{R}_l \end{bmatrix} \tag{8}$$

where $[\mathbf{t}]_\times$ represents the skew-symmetric matrix associated to the translation vector $\mathbf{t}$ and ${}^{d}\mathbf{R}_l$ is a rotation matrix. Furthermore, the interaction matrix $\mathbf{Le}$ is defined as:

$$\mathbf{Le} = \begin{bmatrix} {}^{d}\mathbf{R}_h & \mathbf{0} \\ \mathbf{0} & \mathbf{L}_{\theta\mathbf{u}} \end{bmatrix} \tag{9}$$

where:

$$\mathbf{L}_{\theta\mathbf{u}} = \mathbf{I}_3 - \frac{\theta}{2}[\mathbf{u}]_x + \left(1 - \frac{\text{sinc}\ \theta}{\text{sinc}^2\frac{\theta}{2}}\right) \cdot [\mathbf{u}]_x^2 \tag{10}$$

Finally, to ensure an exponential decoupled decreasing error (i.e. $\dot{\mathbf{e}} = -\lambda \cdot \mathbf{e}$), the control law is defined as:

$$\dot{\mathbf{q}} = -\lambda \cdot \mathbf{J}_s^{\dagger} \cdot \mathbf{e} \tag{11}$$

where λ is the control gain vector and $\mathbf{J}_s^{\dagger}$ the pseudo Moore-Penrose inverse of the feature Jacobian matrix.

In this work, instead of the measured joint angles $\mathbf{q}$ we use the calibrated values of the joint angles $\mathbf{q}+\hat{\boldsymbol{\beta}}$ to compute the robot Jacobians, $\mathbf{J}_h$ and $\mathbf{J}_s$, as well as the calibrated error transformation ${}^{d}\mathbf{T}_h^{[\text{cal}]} = (\mathbf{T}_d)^{-1}\,\mathbf{T}_h^{[\text{cal}]}$.

IV. Results

The markerless visual servoing approach was tested in the iCub robot [1] with an object belonging to the YCB dataset [34] used to benchmark manipulation research. The pudding box used in the experiments is graspable by the iCub robot hand[35]. The object pose is estimated using a fiducial marker - Aruco board [36] exploiting an off-the-shelf implementation provided by the OpenCV library. Then, a fixed roto-translation matrix is defined from the marker (placed on the table) to the object and hand desired poses.

The experiments made in the real platform are reported in this section. First, we define the error metric used to evaluate the experiments. Then, we analyse the robustness of the visual servoing control strategy and the estimation of the hand pose. Finally, we validate the grasping success of the iCub robot comparing both strategies, feedforward (non-calibrated) and feedback (calibrated) approaches.

*a) **Error metrics**:* The validation of the reaching and grasping task relies on measuring the distance between the desired and final positions of the thumb and index fingertips on the object. On the YCB pudding box object used in the experiments, the desired position of the index fingertip is on the lower part of the 'J' of the red JELL-O word printed on the box cover (see Fig. 4); the desired position of the thumb fingertip is at the same height on the other side of the box. We define the distance as:

$$\varphi^{[e]} = \varphi^{[d]} - \varphi^{[f]} \tag{12}$$

where "φ" is either the thumb or the index finger position, $[e]$ denotes the computed error distance, $[d]$ the desired finger position and $[f]$ is the actual final position of the finger. To account for the cases where the object has moved due to hand object collision, we introduce the object displacement measurement (object$^{[e]}$) and define the total error ($[Te]$) for each finger as:

$$\varphi^{[Te]} = \varphi^{[e]} + \varphi^{[e]} \tag{13}$$

*b) **Visual Servoing metrics**:* The evolution of the error pose ($\mathbf{e}$) defined in (5) can be seen in Fig. 2 (a), (b) and (c), respectively the translation component, rotation component and norm. As expected, the error has a smooth exponential

(a) Non Calibrated (b) Calibrated

Fig. 3: Example of an uncalibrated (a) and calibrated (b) eye-hand kinematic chain and the errors in the 2D projection of the thumb, index and middle fingers

decreasing evolution. The step variations at time=20s and time=30s, that can be noted in the orientation error, are caused by a quick and sudden adaptation of the calibration parameters, that in turn generates a new and different estimate of the hand pose. However, these steps in the hand pose estimation are naturally smoothed out by the controller, and thus do not generate discontinuities or sudden variations in the trajectory and speed of the hand movement, as it can be seen in Fig. 2 (d).

*c) **Hand pose estimation**:* The projection of the thumb, index and middle fingers using only the kinematic and the camera models can be seen in Fig. 3 (a); the difference between the real fingers positions in the image and the projected estimation using the kinematic chain can be easily noted. Indeed, these errors can be sufficient to critically increase the reaching error and lead to grasping failures. Instead, Fig. 3 (b) shows the improved hand pose estimation provided by our proposed method, in which the misalignment between the real and estimated hand projection is almost null.

*d) **Grasping task benchmark**:* The robot performed reaching and grasping of the target object in six different locations spread around the workspace. For each one, the robot reaches for the object and tries to grasp it without any visual feedback (i.e. open-loop strategy). Then, it returns to the initial position and restarts the reach-to-grasp movement, performing continuous online calibration of the model and estimation of the hand pose during the motion to refine the alignment between the visual perception and the internal model: this constitutes a markerless position-based visual servoing (i.e. closed-loop strategy). At the end of each reach-to-grasp movement (both open-loop and closed-loop), we measure the distances between the desired and final positions of the thumb and index fingertips.

TABLE I reports the final errors of the fingertips bad their average ($\frac{t+i}{2}$) in the performed experiments. All errors were reduced in the visual servoing (i.e. with online calibration/continuous visual feedback) comparing to the open-loop strategy (i.e. without calibration/visual feedback). Fig. 4 shows the final hand position and the target object in one of the experiments (i.e. "Pose 2"): in the non-calibrated (a) and calibrated (b) case. While in the non-calibrated open-loop

(a) Non Calibrated (b) Calibrated

Fig. 4: Example of the final pose of the hand in one of the experiments: uncalibrated (a) and calibrated (b). In the non-calibrated scenario the grasping fails. The desired position for the index fingertip is the lower part of the J on the cover of the JELL-O Pudding Box.

	Non-Calibrated/Open-loop			**Calibrated/Closed-loop**		
	$\mathrm{t}^{[Te]}$	$\mathrm{i}^{[Te]}$	$\frac{t+i}{2}$	$\mathrm{t}^{[Te]}$	$\mathrm{i}^{[Te]}$	$\frac{t+i}{2}$
Pose 1 [mm]	58.5	39	48.75	19	7.5	13.25
Pose 2 [mm]	75	60	67.5	27	25	26
Pose 3 [mm]	71	49	60	29	22	24.5
Pose 4 [mm]	70	54	62	50	37	43.5
Pose 5 [mm]	56	40	48	41	31	36
Pose 6 [mm]	77	74	75.5	30	19	24.5
Mean	67.92	52.67	60.29	32.67	23.58	27.95
StdDev	8.69	13.20	10.68	11.04	10.19	10.49

TABLE I: Errors measured in the finger tips (t - thumb and i - index) during the experiments. The average errors have decreased by more than a factor of 2.

control case the grasping fails, the proposed markerless visual servoing approach with online hand pose estimation and model calibration allows to perform a successful precision grip of the object. A demonstration of the contribution and performance of this work can be seen in the video attachment (`https://youtu.be/hWb3nFD-xzI`)

In all cases, our online procedure reduces the positioning errors to less than half the non-calibrated case. More importantly, it allows to obtain a very high accuracy in positioning, doing that extra-mile that can permit the execution of precise grips in many practical applications that would be otherwise not accessible. Moreover, it has to be considered that part of the residual positioning error still present in the calibrated case is caused by imprecise estimation of the object pose, that is not a focus of this paper and was realised with a very simple approach.

V. Conclusion and Future Work

We have presented a visual servoing architecture for reaching and grasping tasks in humanoid robots. This is essential for increasing the grasp success rates in non-trivial manipulation scenarios and to cope with unavoidable uncertainties and calibration errors of robots with complex kinematics chains and "eye-to-hand" visual configurations. The method was implemented in the iCub robot and experi-

ments were made showing its effectiveness and benefits with respect to the conventional open-loop approach implemented in most of the current works. With the presented work we have demonstrated the feasibility of visual servoing in humanoid robots without markers and hope to spawn further research on the application of visual feedback in the control of manipulation actions, paving the way for truly adaptive systems, able to react to disturbances and errors both in the environment and in the robot model itself. Our future work will focus on combining the presented approach with image based visual servoing methods, to compensate also for possible errors arising in the estimation of the object's pose.

ACKNOWLEDGMENTS

This work was partially supported by Fundação para a Ciência e a Tecnologia [UID/EEA/50009/2013]. Pedro Vicente is funded by a PhD grant from Instituto de Sistemas e Robótica (ISR), Instituto Superior Técnico, and Universidade de Lisboa.

REFERENCES

[1] G. Metta, L. Natale, F. Nori, G. Sandini, D. Vernon, L. Fadiga, C. von Hofsten, K. Rosander, M. Lopes, J. Santos-Victor, A. Bernardino, and L. Montesano, "The icub humanoid robot: an open-systems platform for research in cognitive development," *Neural Netw.*, vol. 23, 2010.

[2] Chaumette and S. Hutchinson, "Visual servo control, part i: Basic approaches," *IEEE Robot. and Autom. Mag.*, vol. 13, pp. 82–90, 2006.

[3] J. Bohg, A. Morales, T. Asfour, and D. Kragic, "Data-driven grasp synthesis - a survey," *IEEE Trans. Robot.*, 2014.

[4] P. Vicente, L. Jamone, and A. Bernardino, "Robotic Hand Pose Estimation Based on Stereo Vision and GPU-enabled Internal Graphical Simulation," *J. Intell. & Robotic Syst.*, pp. 1–20, 2016.

[5] ——, "Online body schema adaptation based on internal mental simulation and multisensory feedback," *Frontiers in Robotics and AI*, vol. 3, no. 7, 2016.

[6] R. Figueiredo, A. Shukla, D. Aragao, P. Moreno, A. Bernardino, J. Santos-Victor, and A. Billard, "Reaching and grasping kitchenware objects," in *Proc. Int. Symp. Syst. Integration (SII)*, 2012.

[7] A. S. Seungsu Kim and A. Billard, "Catching objects in flight," *IEEE Trans. Robot.*, 2014.

[8] J. Stckler, M. Schwarz, M. Schadler, A. Topalidou-Kyniazopoulou, and S. Behnke, "Nimbro explorer: Semiautonomous exploration and mobile manipulation in rough terrain," *J. Field Robotics*, vol. 33, no. 4, pp. 411–430, 2015.

[9] D. Leidner, W. Bejjani, A. Albu-Schaeffer, and M. Beetz, "Robotic agents representing, reasoning, and executing wiping tasks for daily household chores," in *Proc. Int. Conf. Autonomous Agents & Multiagent Systems (AAMAS)*, 2016, pp. 1006–1014.

[10] D. Leidner, A. Dietrich, M. Beetz, and A. Albu-Schäffer, "Knowledge-enabled parameterization of whole-body control strategies for compliant service robots," *Auton. Robots*, vol. 40, no. 3, pp. 519–536, 2016.

[11] T. La Anh and J.-B. Song, "Robotic grasping based on efficient tracking and visual servoing using local feature descriptors," *Int. J. Precision Engineering and Manufacturing*, vol. 13, no. 3, pp. 387–393, 2012.

[12] G. Ma, Q. Huang, Z. Yu, X. Chen, L. Meng, M. Sultan, W. Zhang, and Y. Liu, "Hand-eye servo and flexible control of an anthropomorphic arm," in *IEEE Int. Conf. on Robot. and Biomimetics (ROBIO)*, Dec 2013, pp. 1432–1437.

[13] R. Barth, J. Hemming, and E. J. van Henten, "Design of an eye-in-hand sensing and servo control framework for harvesting robotics in dense vegetation," *Biosystems Engineering*, vol. 146, pp. 71–84, 2016.

[14] C. Kulpate, R. Paranjape, and M. Mehrandezh, "Precise 3d positioning of a robotic arm using a single camera and a flat mirror," *Int. J. of Optomechatronics*, vol. 2, no. 3, pp. 205–232, 2008.

[15] O. Birbach, U. Frese, and B. Ba, "Rapid calibration of a multi-sensorial humanoid's upper body : An automatic and self-contained approach," *Int. J. Robotics Research*, vol. 34, no. 4-5, pp. 420–436, 2015.

[16] N. Vahrenkamp, S. Wieland, P. Azad, D. Gonzalez, T. Asfour, and R. Dillmann, "Visual servoing for humanoid grasping and manipulation tasks," in *IEEE-RAS Int. Conf. Humanoid Robots*, Dec 2008, pp. 406–412.

[17] N. Vahrenkamp and T. Asfour, "Representing the robot's workspace through constrained manipulability analysis," *Auton. Robots*, vol. 38, no. 1, pp. 17–30, 2015.

[18] L. Jamone, L. Natale, F. Nori, G. Metta, and G. Sandini, "Autonomous online learning of reaching behavior in a humanoid robot," *Int. J. Humanoid Robotics*, vol. 09, no. 03, p. 1250017, 2012.

[19] L. Jamone, M. Brandao, L. Natale, K. Hashimoto, G. Sandini, and A. Takanishi, "Autonomous online generation of a motor representation of the workspace for intelligent whole-body reaching," *Robot. Auton. Syst.*, vol. 62, no. 4, pp. 556 – 567, 2014.

[20] L. Jamone, B. Damas, N. Endo, J. Santos-Victor, and A. Takanishi, "Incremental development of multiple tool models for robotic reaching through autonomous exploration," *Paladyn J. Behavioral Robotics*, vol. 3, no. 3, pp. 1–15, 2012.

[21] D. Agravante, J. Pages, and F. Chaumette, "Visual servoing for the reem humanoid robot's upper body," in *IEEE Int. Conf. Robotics and Automation (ICRA)*, May 2013, pp. 5253–5258.

[22] S. Koo and S. Behnke, "Focused online visual-motor coordination for a dual-arm robot manipulator," in *IEEE Int. Conf. Robotics and Automation (ICRA)*, May 2016, pp. 1579–1586.

[23] F. Widmaier, D. Kappler, S. Schaal, and J. Bohg, "Robot arm pose estimation by pixel-wise regression of joint angles," in *IEEE Int. Conf. Robotics and Automation (ICRA)*, May 2016, pp. 616–623.

[24] X. Gratal, J. Romero, and D. Kragic, "Virtual visual servoing for real-time robot pose estimation," in *Proc. IFAC World Congress*, 2011, pp. 9017–9022.

[25] A. I. Comport, E. Marchand, M. Pressigout, and F. Chaumette, "Real-time markerless tracking for augmented reality: the virtual visual servoing framework," *IEEE Trans. Visualization and Computer Graphics*, vol. 12, no. 4, pp. 615–628, July 2006.

[26] X. Gratal, C. Smith, M. Bjrkman, and D. Kragic, "Integrating 3d features and virtual visual servoing for hand-eye and humanoid robot pose estimation," in *IEEE-RAS Int. Conf. Humanoid Robots*, Oct 2013, pp. 240–245.

[27] S. M. Lavalle, "Rapidly-exploring random trees: A new tool for path planning," Computer Science Dept., Iowa State University, Tech. Rep., 1998.

[28] V. Lepetit and P. Fua, "Monocular model-based 3d tracking of rigid objects," *Found. Trends. Comput. Graph. Vis.*, vol. 1, no. 1, pp. 1–89, Jan. 2005.

[29] N. Kantas, A. Doucet, S. Singh, and J. Maciejowski, "An overview of sequential monte carlo methods for parameter estimation on general state space models," in *IFAC Symp. System Identification (SYSID)*, 2009, pp. 774–785.

[30] G. Borgefors, "Distance transformations in digital images," *Computer Vision, Graphics and Image Processing*, vol. 34, no. 3, pp. 344–371, Jun. 1986.

[31] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 8, no. 6, pp. 679–698, Nov 1986.

[32] G. Borgefors, "Hierarchical chamfer matching: a parametric edge matching algorithm," *IEEE Trans. Pattern Anal. Mach. Intell*, vol. 10, no. 6, pp. 849–865, 1988.

[33] B. Siciliano and O. Khatib, *Springer Handbook of Robotics*. Springer-Verlag New York, Inc., 2007.

[34] B. Calli, A. Singh, A. Walsman, S. Srinivasa, P. Abbeel, and A. M. Dollar, "The YCB object and Model set: Towards common benchmarks for manipulation research," in *Int. Conf. Advanced Robot (ICAR)*, July 2015, pp. 510–517.

[35] L. Jamone, A. Bernardino, and J. Santos-Victor, "Benchmarking the grasping capabilities of the icub hand with the ycb object and model set," *IEEE Robot. and Autom. Letters*, vol. 1, no. 1, pp. 288–294, Jan 2016.

[36] S. Garrido-Jurado, R. Munoz-Salinas, F. Madrid-Cuevas, and M. Martin-Jimenez, "Automatic generation and detection of highly reliable fiducial markers under occlusion," *Pattern Recognition*, vol. 47, no. 6, pp. 2280 – 2292, 2014.