

# Error Regulation Strategies for Model Based Visual Servoing Tasks

## Application to Autonomous Object Grasping with Nao Robot

Amine Abou Moughlbay

Institut de Recherche en Communications  
et Cybernétique de Nantes (IRCCyN)  
Ecole Centrale de Nantes - France  
Amine.abou-moughlbay@ec-nantes.fr

Enric Cervera

Robotic Intelligence Lab  
Jaume-I University  
Castello-Spain  
ecervera@icc.uji.es

Philippe Martinet

IRCCyN - Ecole Centrale de Nantes  
Institut Pascal - Clermont Ferrand  
France  
Philippe.Martinet@irccyn.ec-nantes.fr

**Abstract**—When applying service robotic tasks using sensor based control, a classical exponential decrease of the error is usually used in the control laws which can reduce the performance of the executed task. In fact, due to this choice, the convergence time greatly increases especially at the end of the process. To ameliorate the performance of such tasks, we present in this paper two new error regulation strategies to accelerate the service tasks execution. These propositions are compared with the classical one in the case of performing autonomous object's manipulation tasks using real-time visual servoing. The Model Based Tracking method is used to apply head servoing and grasping of different objects using Nao humanoid robot.

### I. INTRODUCTION

When applying manipulation tasks, a key issue to be considered is the study of the interaction between the robot and its environment. Indeed, the control of the position, velocity or forces applied on the contact points is essential to perform the desired tasks. In uncertain environments, robot's end-effectors motion may be subject to online modifications to accommodate unexpected events or to respond to sensor inputs. When controlling the interaction between the manipulator and the environment, the desired task is often specified in the Operational Space and requires precise control of the end-effector motion, Joint Space Control schemes are not suitable in these situations. Thus programming of service and manipulation tasks is most conveniently accomplished by directly specifying data at the contact points, rather than specifying the joint positions and velocities required to achieve them [1].

The use of sensor based control for the robotic service tasks is very common and was applied in several developments these last years: different works have been carried out in teleoperation area, by controlling a robotic system to perform manipulation tasks at a distance using a multi-modal human-system interface. It was also applied to bi-manual manipulation while walking [2], dexterous telemanipulation [3] and in space teleoperation [4].

Furthermore, many practical learning control systems are used to control complex robotic systems involving multiple

feedback sensors and multiple command variables during both repetitive and nonrepetitive operations [5]. The issue of teaching a robot to manipulate everyday objects through human demonstration has been studied by [6] who proposed a method that enables a robot to decompose a demonstrated task into sequential manipulation primitives, series of sequential rotations and translations [7].

Among all feedbacks used in sensor based control, the visual information provides the most important and instant cues for perception of the interaction with the working environment. Compared to already mentioned methods, 3D visual servoing provides very efficient solutions to control robot motions. It supplies high positioning accuracy, good robustness to sensor noise and calibration uncertainties, and reactivity to environment changes [8][9].

In this paper, only Sensor Based Control formalism is used to perform the desired tasks. More precisely, 3D visual feedback data and Model Based Tracking (MBT) techniques are used to execute, in real-time and closed loop, manipulation tasks on the humanoid mobile robot Nao. The main goal is to execute these tasks as fast as possible, thus we propose to improve the task performance by using two new error regulation strategies instead of the classical exponential decrease of the error. In the second section, we present the used visual servoing control law and the different error regulations strategies. In section III, we introduce the case study platform: the Nao robot and the elementary tasks employed to apply object's manipulation. A comparison of the experimental results given by the proposed strategies is presented in section IV. The final section discusses conclusions and draws future works.

### II. SYSTEM ARCHITECTURE

#### A. Visual servoing control law

A large variety of positioning or target tracking tasks can be implemented by controlling from one to all DOF of the system. Whatever the sensor configuration, which can vary from one camera mounted on the robot end-effector to several

free-standing cameras, a set of visual features ( $\mathbf{s}$ ) has to be designed from the visual measurements obtained from the system configuration  $\mathbf{x}(t)$  allowing control of the desired DOF.

In the case of a motionless environment, the signal time derivative ( $\dot{\mathbf{s}}$ ) is directly related to the sensor velocity screw ( $\mathbf{V}_c$ ) expressed in the sensor frame:

$$\dot{\mathbf{s}} = \mathbf{L}_s \mathbf{V}_c \quad (1)$$

where  $\mathbf{L}_s$  is named the interaction matrix of  $\mathbf{s}$ . Its analytical form can be derived for many features coming from exteroceptive sensors. It depends mainly on the type of considered sensory data  $\mathbf{s}$  and on the sensor intrinsic parameters.

A control law is thus designed so that these features  $\mathbf{s}$  reach a desired value  $\mathbf{s}^*$ , defining a correct realization of the task. Indeed, in visual servoing, for a desired decrease  $\dot{\mathbf{e}}$  of the error  $\mathbf{e} = (\mathbf{s} - \mathbf{s}^*)$ , the camera velocity is considered as input of the robot controller and given by:

$$\mathbf{V}_c = \mathbf{L}_s^+ \dot{\mathbf{e}} \quad (2)$$

where  $\mathbf{L}_s^+$  is the pseudo-inverse of the interaction matrix (for more details refer to [10]).

Using the kinematic matrix  $\mathbf{J}$  which relates camera velocity  $\mathbf{V}_c$  with robot's joints velocity ( $\mathbf{V}_c = \mathbf{J} \dot{\mathbf{q}}$ ), the general control law used to define a task will be given by:

$$\dot{\mathbf{q}} = (\mathbf{L}_s \mathbf{J})^+ \dot{\mathbf{e}} \quad (3)$$

## B. Error regulation strategies

### 1) Classical exponential decrease:

When executing service tasks, and especially object's manipulation, the main goal is to carry out these tasks precisely and as fast as possible. Thus to decrease the task error rapidly and to arrive to an acceptable precision. Classically, an exponential decrease (4) is used to decrease the error when applying this control law:

$$\dot{\mathbf{e}} = -\lambda \mathbf{e} \quad (4)$$

with  $\lambda$  a proportional gain that is usually tuned to minimize the time to convergence.

In this case, the error follows an asymptotic exponential decrease to zero. But this choice leads to a larger regulation time (red curve in Fig.1). Furthermore, this gain should be tuned to reduce the convergence rate of the main task while preserving the stability of the system [11] and reducing error oscillations near convergence.

Moreover, when this classical regulation is applied into visual servoing tasks, and especially object's tracking, an increase of the gain  $\lambda$  may lead to tracking loss due to large camera velocities, or robot's joint velocity saturation. To avoid these issues, a small gain value is generally used to initialize

the tracking and maintained constant till the end of the task. But in the other hand, this consideration leads to an increase of the convergence time.

Indeed near convergence, the displacement of the robot's camera and control points become smaller, thus a larger gain may be used to ameliorate task performance. From that, a first proposal is to increase the value of  $\lambda$  after task initialization. Note that the gain used initially is optimal for the task initialization and cannot be increased from the beginning. Actually the use of a higher value leads to tracking crash and task failure due to the fast motions.

### 2) First proposition of error regulation:

The goal is thus to decrease the convergence time of the control law to apply manipulations tasks rapidly without loosing the system stability.

Thus we introduced a varying reduction of the error (5) which acts as an exponential decrease (with a gain  $\lambda_0$ ) from initial value ( $\mathbf{e}_0$ ) until arriving to a specified threshold ( $\mathbf{e}_{lim}$ ) near the equilibrium where it switches to a faster exponential decrease of the error (with a gain  $\lambda_1 > \lambda_0$ ) to carry out the task as fast as possible and to arrive to the desired precision ( $\mathbf{e}_{inf}$ ) which indicates the task accomplishment (blue curve in Fig.1).

The used decrease of the error is thus given by:

$$\dot{\mathbf{e}} = \begin{cases} -\lambda_0 \mathbf{e} & \text{for } \|\mathbf{e}\| \geq \mathbf{e}_{lim} \\ -\lambda_1 \mathbf{e} & \text{for } \|\mathbf{e}\| < \mathbf{e}_{lim} \end{cases} \quad (5)$$

As presented in Fig.1, this function is continuous and allows a decrease of the convergence time from  $t_f$  with the classical exponential decrease given in (4) to  $t_1$  with this method.

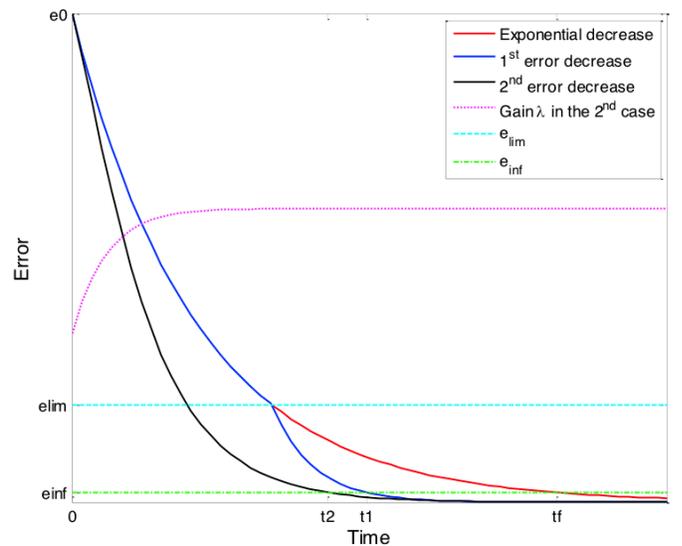


Fig. 1: Comparison between the proposed strategies of error regulation

The improvement in time owned to this method, in function of the initial ( $\lambda_0$ ) and final ( $\lambda_1$ ) exponential gains, initial values of error  $e_0$ , desired moment of switching  $e_{lim}$  and the desired precision  $e_{inf}$ , can be calculated theoretically by:

$$\frac{t_1}{t_f} = \frac{\lambda_0}{\lambda_1} + \left(1 - \frac{\lambda_0}{\lambda_1}\right) \frac{\log\left(\frac{e_0}{e_{lim}}\right)}{\log\left(\frac{e_0}{e_{inf}}\right)} \quad (6)$$

### 3) Second proposition of error regulation:

To avoid discontinuity in error regulation velocity and possible task instability due to the switch from a low gain value to a higher one, another formulation of the error regulation can be used to have a smoother curve and to further decrease the time to convergence.

The formulation given by (7) uses the error norm to increase the gain value when the error decreases:

$$\dot{e} = -\lambda e \quad \text{with} \quad \lambda = \lambda_0 + \mathbf{a} \exp(-\mathbf{b} \|e\|) \quad (7)$$

where  $\mathbf{a}$ ,  $\mathbf{b}$  and  $\lambda_0$  are positive constant scalar values.

On the contrary of the classical method where the gain value is constant during error regulation, in this case the value of  $\lambda$  begins with a small value, to ensure task initialization and stability, and then increases when the error norm arrives near zero. The gain variation for this proposition is illustrated by the magenta curve in Fig.1.

The representation of the corresponding error regulation (black curve) shows that the error variation function is continuous and decreases the convergence time from  $t_f$  with the classical exponential decrease to  $t_2$  (depending on the choice of the parameters).

## III. TASKS IDENTIFICATION

In this section, we introduce briefly the system architecture of the Nao robot and we present the generic tasks which are used in the experimental part to apply object's manipulation, and consequently to compare the proposed error regulation strategies.

### A. Nao Architecture

Nao Robot [12], developed by Aldebaran robotics, is a biped robot with 25 Degrees of Freedom (DOF). It has 3-fingered robotic hands used for grasping and holding small objects (it can carry up to 300g using both hands). It is equipped with: 2 ultrasound devices situated in the chest that provide space information in 1 meter range distance, 2 cameras situated on the top and bottom of the head, 2 bumpers (contact sensors on the robot's feet), a gyrometer and an accelerometer (to determine whether the robot is in a stable or unstable position).

To execute the different tasks, we should define frames on the robot's body and environment's items (see Fig. 2). In Nao's body we consider the following frames: Nao's space frame

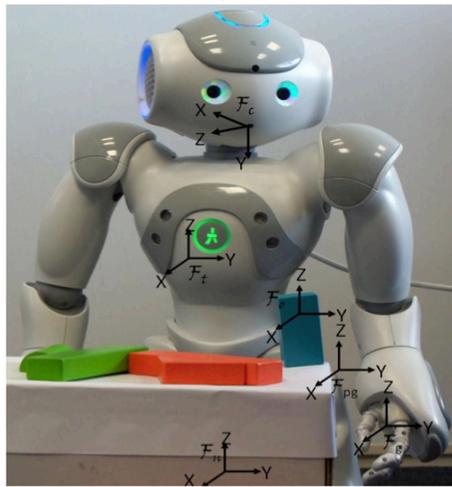


Fig. 2: Useful frames in Nao's environment

$\mathcal{F}_n$  (between robot's feet),  $\mathcal{F}_t$  on robot's torso,  $\mathcal{F}_c$  a camera attached frame, robot's hand frame  $\mathcal{F}_h$ , robot's gripper frame  $\mathcal{F}_g$  and a pre-grasping frame  $\mathcal{F}_{pg}$ . In the robot's environment, we define the object's frame  $\mathcal{F}_o$ .

### B. Tracking technique

Many tracking tools have been implemented in several visual servoing toolboxes. On the Visual Servoing Platform (ViSP) [13], we find a dot tracker, a moving edges tracker, and a 3D model-based tracker. The last one [14] provides a robust solution to track geometrical shapes (lines, cylinders, ellipsoids, ...) as soon their perspective projection can be computed. It estimates online the position of a known object in the camera frame.

This method consists of locally tracking 2D contour points, and to estimate model's pose based on a non linear iterative algorithm using a virtual visual servoing technique. It requires a 3D model and needs to compute the initial pose which is used to project the model on the image. The tracking method assumes that the pose corresponding to the previous image is known, the new lines are tracked, and the goal is to move the pose to match the object in the new image with the projection of the model. The error function ( $err$ ) between image features  $p_i$  and model projection  $q_i$  is thus minimized along the normal direction  $\vec{n}$  (see Fig. 3):

$$err = \sum_i \Delta(p_i, q_i) = \sum_i |(q_i - p_i) \cdot (n_i)| \quad (8)$$

### C. Generic Tasks

While Nao is a capable platform; due to the complexity of the problem to be solved, the capabilities of the robot can be enhanced and the complexity of the problem can be reduced by decomposing it in simpler generic tasks:

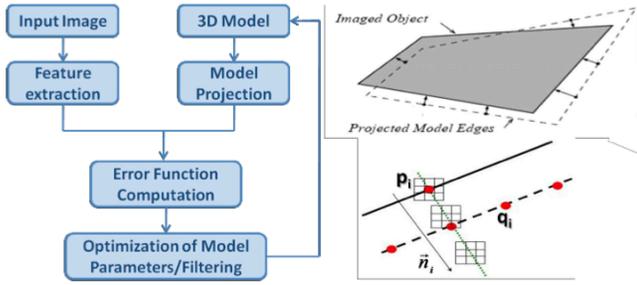


Fig. 3: Model Based tracking system using the Moving edge detection.

### 1) Detection and Tracking Tasks:

Using the MBT technique of ViSP, we initialize the tracker manually and determine in real-time the pose (position/orientation) of the desired item to manipulate. An automatic re-initialization of the tracker is implemented: it uses the last poses of the tracked object to be reinitialized in case of a failure due to an occlusion or the fast motions of the robot's camera. Thus the model is automatically detected and tracked; this task allows us to determine instantly the pose of the desired item frame in the robot camera's frame in form of a homogeneous transformation matrix ( ${}^cM_o$ ).

### 2) Visibility Task:

This task consists of controlling the robot's head pose to focus a (fixed/mobile) point of the environment (item's center, gripper, virtual point...) in the center of the camera's image. Throughout this application on Nao robot, the visibility task is used for controlling the orientation of the head to focus the object's center in the center of the camera's image. 2 DOF are used by this task to control the head's Yaw and Pitch. The task's goal is thus to regulate the horizontal and vertical position of the center of the object projection  $s_{x,y} = {}^cT_{o(x,y)}$  to zero ( $s^* = (0, 0)$ ).

Using the object 3D pose  ${}^cT_o = (X, Y, Z)^T$ , and the 2D pose  $(x, y)^T$  of the tracked point (projection of 3D point in the normal image plane), we apply the control law defined in (3) using the visual primitive  $s = (x, y)$  and its corresponding interaction matrix  $L_s$  given by:

$$L_s = \begin{bmatrix} -\frac{1}{Z} & 0 & \frac{x}{Z} & xy & -(1+y^2) & y \\ 0 & -\frac{1}{Z} & \frac{y}{Z} & 1+x^2 & -xy & -x \end{bmatrix} \quad (9)$$

Note that for this task the kinematic matrix  $J$  required in the control law (3) uses the Jacobian of the robot's head control point calculated from the robot's geometric model.

### 3) Pre-Grasping and Grasping Tasks:

This task uses the hand's control point and allows the robot to move it to a desired static/mobile pose. It can be used to perform the pre-grasping, grasping, and displacing objects tasks.

In case of pre-grasping task the goal pose ( ${}^gM_{pg}$ ) is determined using the grasping strategies. These strategies depend on the geometry of the object to manipulate and the geometry of the robot's gripper. The grasping strategy controls the relative position and/or angle between the gripper and the item to grasp [15].

According to Nao's gripper's geometry (of one DOF) and the item's shape (rectangular model), 4 DOF are enough to execute this task: 3 DOF constraints the gripper's pose and 1 DOF (yaw angle) for the gripper's orientation. The task's target is then to move the robot's arm to the pre-grasping pose. The task's error is extracted from the relative pose between the gripper and pre-grasping point ( ${}^gM_{pg}$ ) which is regulated to zero.

In case of a grasping task, the same technique is considered and the same number of DOF is constrained as in the previous case of pre-grasping, except the desired gripper pose which is changed to the object's pose. Thus the task's error will be extracted from the relative pose between the gripper and the item  ${}^gM_o$  which is also regulated to zero [16].

Considering that the visual primitive is parameterized by  $s = (\mathbf{t}, \mathbf{u}\theta)$  where  $\mathbf{t}$  is the position error between the current and desired frame, while  $\mathbf{u}\theta$  is the orientation error, decomposed as the axis  $\mathbf{u}$  and angle  $\theta$  of the rotation between these two frames. The control law (3) is then applied using the Jacobian at the robot's gripper and the corresponding interaction matrix  $L_s$  given by:

$$L_s = \begin{bmatrix} -\mathbf{I}_3 & [\mathbf{t}]_{\times} \\ \mathbf{0}_3 & L_{\omega} \end{bmatrix} \quad (10)$$

where  $\mathbf{I}_3$  and  $\mathbf{0}_3$  are the  $3 \times 3$  identity and zero matrices respectively,  $[\mathbf{t}]_{\times}$  is the skew symmetric matrix associated with vector  $\mathbf{t}$ , and  $L_{\omega} = \mathbf{I}_3 - \frac{\theta}{2}[\mathbf{u}]_{\times} + \left(1 - \frac{\text{sinc}(\theta)}{\text{sinc}^2(\theta/2)}\right)[\mathbf{u}]_{\times}^2$ .

## IV. EXPERIMENTAL RESULTS

The presented tasks in the previous section have been implemented and tested on the Humanoid Nao robot with a control rate equal to that of the camera (20 Hz). To visualize the effect of the varying decrease of the error, presented previously, these tasks are executed on parallel using the three propositions of error decrease given in (4), (5) and (7). The initial conditions and parameters used for each method are given in Table I.

Experiment photos of the grasping tasks executed by Nao robot are presented in Fig. 4 and correspond to the tasks presented in section III-C above. The first one, Fig. 4-a, shows the item to grasp deposited on the table before launching the MBT to detect and track it (Fig. 4-b). The visibility task is used to center the object on the camera's image, and simultaneously the pre-grasping task is executed in Fig. 4-c, where we can identify the different frames on the robot's arm and gripper in addition of the object's frame. Afterwards, the gripper's frame approaches the object's one when executing

Parameters	Visibility Task	Pre-Grasping Task	Grasping Task
$e_0$	0.3 m	0.25 m 180 deg	0.005 m 15 deg
$e_{inf}$	0.005 m	0.005 m 15 deg	0.001 m 3 deg
$e_{lim}$	0.05 m	0.05 m	0.005 m
$\lambda, \lambda_0$	0.02	1.2	1.2

TABLE I: Initial conditions and parameters values

the grasping task (Fig. 4-d). Finally the gripper closes and the manipulation task is completed (Fig. 4-e-f).

### A. Exponential regulation of the error

In this part, we present experimental results using the control law (3) and the exponential decrease of the error given by (4). The regulation of the error in each task is presented in Fig.5a: the first graph represents the horizontal and vertical position error during the head servoing task (visibility task), initially the object is on a distance of approximately 300 mm from the center of the camera's image, we remark that this error is successfully regulated exponentially to zero during almost 29 sec with a precision of 5 mm.

For the 2<sup>nd</sup> and 3<sup>rd</sup> graphs, we present the pre-grasping and grasping tasks errors on X, Y and Z components (in Nao's frame), and the Yaw angle of the gripper orientation. During these tasks, they are regulated exponentially to the desired precision during 31 sec.

### B. First proposition of error regulation

In Fig.5b, we used the control law (3) with the first proposition of error regulation given by (5) with the same initial conditions and parameters (Table I). The switching between the two modes is executed when arriving to  $e_{lim}$ . Referring to the error regulation curves, the visibility task is executed in 21.7 sec and the grasping tasks in 18.6 sec with a 40 % time improvement with respect to the previous method. This result is a bit different from the theoretical one using the relation (6) which gives an improvement of 45 % due to experimental reasons.

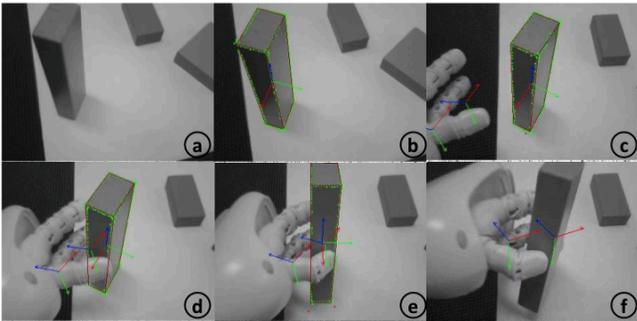
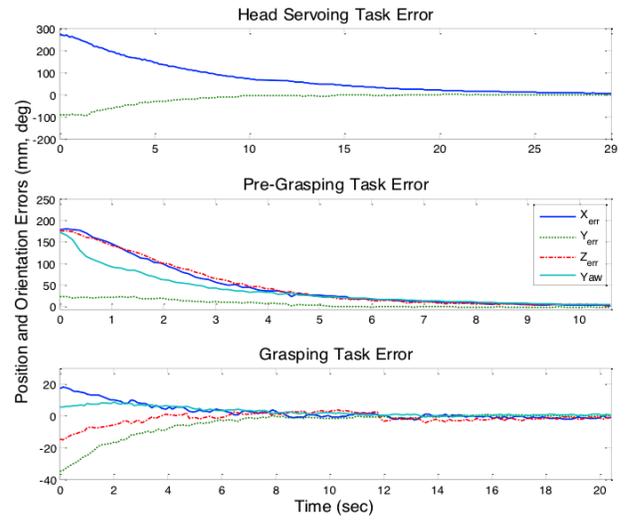
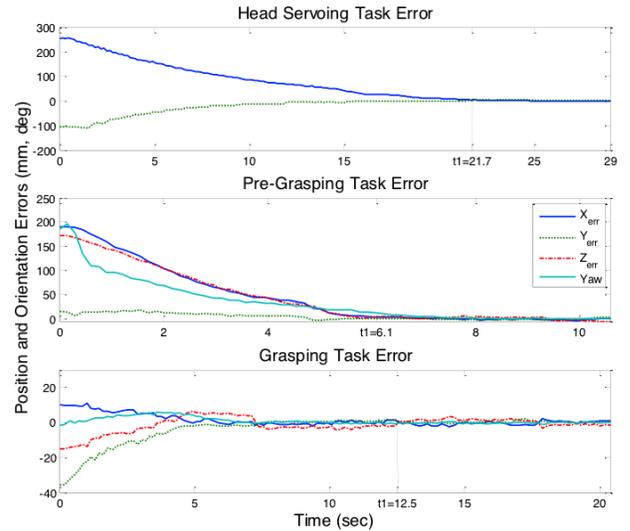


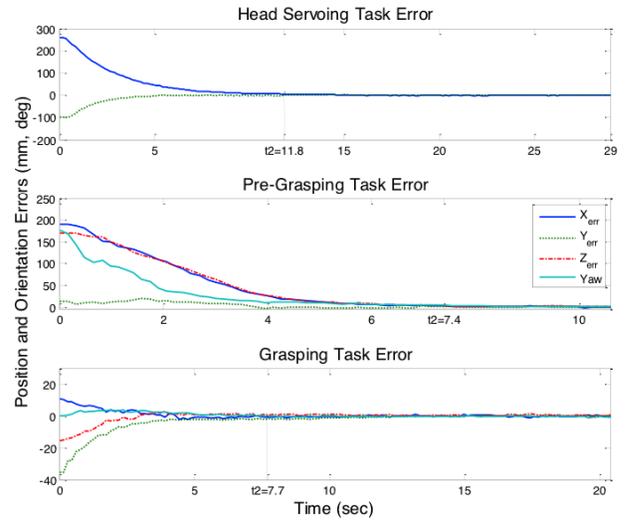
Fig. 4: Experiment Photos of tracking and grasping of an item showing the arm's, gripper's and object's frames



(a) Case of exponential decrease of the error



(b) Case of the first strategy of error regulation



(c) Case of the second strategy of error regulation

Fig. 5: Experimental results for the three methods

Method	Visibility Task	Pre-Grasping Task	Grasping Task
Case (4)	26.3 sec	10.96 sec	17.41 sec
Case (5)	15.56 sec 40.1 %	5.43 sec 50.4 %	9.06 sec 48 %
Case (7)	11.74 sec 55.4 %	6.41 sec 41.5 %	8.26 sec 52.6 %

TABLE II: Average convergence time (in seconds) and improvement (in percent) over the classical method

### C. Second proposition of error regulation

The control law (3) is used with the second proposition of error regulation given by (7). Referring to the variation of the error in each task, presented in Fig.5c, the visibility task is executed in 11.8 sec with a 59 % time improvement over the classical one. The pre-grasping and grasping tasks are executed in 15.1 sec with a 51 % time improvement.

### D. Discussion

To ensure the feasibility and efficiency of the proposed error regulation strategies, and their improvement in convergence time. The previously presented experiments have been repeated 40 times (for each method) with the same initial conditions and we calculated the average of the required convergence time in each case and the improvement over classical one (Results are represented in Table II).

Referring to the results of the three strategies and the represented graphs (Fig.5), we remark that the manipulation task is executed faster with a good improvement of the convergence time between 40 % and 55 %. Furthermore, the system is stable after finishing the desired tasks and not perturbed by the defined varying gains.

By comparing the defined gains for each task, we remarks that in the first proposition (5) the gain begins with a small value  $\lambda_0$  than increases to  $\lambda_1$ , otherwise in this case (7) the gain begins with a relatively high value and increases all the time until stabilizing when arriving to a small error value (magenta curve in Fig.1). Thus the first method (5) is good for the visibility task because of the low gain at the beginning of the tracking but the switch to high gain may leads to some oscillations when convergence of the grasping task. Otherwise, a good behavior is remarked near convergence for the second proposition but it begins with a relatively high gain which may influence on the tracking initialization due to high camera's motion, furthermore parameter tuning is very necessary for avoiding oscillations or instability.

Furthermore, the represented graphs shows that the MBT is ensured to be efficient for robot's tracking and grasping tasks. These results shows also that this method is robust to camera occlusion by the robot's hand, and robust to small object movements due to hand-object collision.

## V. CONCLUSIONS

In this paper, we studied the use of the real-time visual servoing techniques to perform object's manipulation tasks.

Indeed, the Model Based Tracking method is used to apply, in parallel, head servoing and grasping of different objects. To accomplish the task as fast as possible, two propositions of error regulation are introduced and compared with the exponential decrease used in classical control law. Repeated experiments on the Nao robot have been executed to ensure the efficiency of the proposed methods.

Future works will concentrate on the extension of these regulation strategies from the autonomous object grasping to other service robotic tasks. Moreover, their implementation on other complex platforms with different objects should be accomplished to ensure the robustness of this method. For the experimental part, other sensor's feedbacks should be used to improve the manipulation reactivity against dynamic or unusual changes in the environment, especially after grasping the object where many occlusions and brutal motions appears.

## REFERENCES

- [1] O. Brock, J. Kuffner, and J. Xiao, "Motion for manipulation tasks," in *Springer Handbook of Robotics*, B. Siciliano and O. Khatib, Eds. Springer Berlin Heidelberg, 2008, pp. 615–645.
- [2] N. Mansard, O. Stasse, F. Chaumette, and K. Yokoi, "Visually-guided grasping while walking on a humanoid robot," in *2007 IEEE Int. Conf. on Robotics and Automation*, apr. 2007, pp. 3041 –3047.
- [3] W. Griffin, W. Provancher, and M. Cutkosky, "Feedback strategies for shared control in dexterous telemanipulation," in *2003 IEEE Int. Conf. on Intelligent Robots and Systems*, vol. 3, oct. 2003, pp. 2791 – 2796.
- [4] N. Lii, Z. C., B. Pleintinger, C. Borst, G. Hirzinger, and A. Schiele, "Toward understanding the effects of visual- and force-feedback on robotic hand grasping performance for space teleoperation," in *IEEE Int. Conf. on Intelligent Robots and Systems*, oct. 2010, pp. 3745 –3752.
- [5] W. T. Miller, "Sensor-based control of robotic manipulators using a general learning algorithm," *IEEE Journal of Robotics and Automation*, vol. 3, no. 2, pp. 157–165, apr. 1987.
- [6] H. Dang and P. Allen, "Robot learning of everyday object manipulations via human demonstration," in *2010 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems - IROS'10*, oct. 2010, pp. 1284 –1289.
- [7] A. Jain and C. Kemp, "Pulling open novel doors and drawers with equilibrium point control," in *the 9th IEEE-RAS Int. Conf. on Humanoid Robots*, dec. 2009, pp. 498 –505.
- [8] W. Wilson, C. Williams Hulls, and G. Bell, "Relative end-effector control using cartesian position based visual servoing," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 5, pp. 684 –696, oct 1996.
- [9] B. Thuilot, P. Martinet, L. Cordesses, and J. Gallice, "Position based visual servoing: keeping the object in the field of vision," in *2002 IEEE Int. Conf. on Robotics and Automation*, vol. 2, 2002.
- [10] F. Chaumette and S. Hutchinson, "Visual servo control. i. basic approaches," *IEEE Robotics Automation Magazine*, vol. 13, no. 4, pp. 82 –90, dec. 2006.
- [11] M. Marey and F. Chaumette, "A new large projection operator for the redundancy framework," in *2010 IEEE Int. Conf. on Robotics and Automation*, may 2010, pp. 3727 –3732.
- [12] D. Gouaillier, V. Hugel, P. Blazevic, C. Kilner, J. Monceaux, P. Lafourcade, B. Marnier, J. Serre, and B. Maisonnier, "Mechatronic design of nao humanoid," in *2009 IEEE Int. Conf. on Robotics and Automation*, may 2009, pp. 769 –774.
- [13] E. Marchand, F. Spindler, and F. Chaumette, "Visp for visual servoing: a generic software platform with a wide class of robot control skills," *IEEE Robotics Automation Magazine*, vol. 12, pp. 40 –52, dec. 2005.
- [14] A. Comport, E. Marchand, M. Pressigout, and F. Chaumette, "Real-time markerless tracking for augmented reality: the virtual visual servoing framework," *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 4, pp. 615 –628, jul-aug. 2006.
- [15] J. Sorribes, M. Prats, and A. Morales, "Visual tracking of a jaw gripper based on articulated 3d models for grasping," in *2010 IEEE Int. Conf. on Robotics and Automation*, may 2010, pp. 2302 –2307.
- [16] P. Martinet and J. Gallice, "Position based visual servoing using a non-linear approach," in *1999 IEEE Int. Conf. on Intelligent Robots and Systems*, vol. 1, 1999, pp. 531 –536 vol.1.