Improving mutual information based visual servoing

Amaury Dame, E. Marchand

▶ **To cite this version:**

Amaury Dame, E. Marchand. Improving mutual information based visual servoing. IEEE Int. Conf. on Robotics and Automation, ICRA'10, 2010, Anchorage, Alaska, United States. pp.5531-5536. inria-00544785

HAL Id: inria-00544785
https://inria.hal.science/inria-00544785

Submitted on 9 Dec 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Improving mutual information-based visual servoing

Amaury Dame, Eric Marchand

Abstract— **In a previous paper [3], we proposed a new way to achieve visual servoing. Rather than minimizing the error between the position of two set of geometric features, we proposed to maximize the mutual information shared by the current and desired images. This leads to a new information theoretic approach to visual servoing. Mutual information is a well known alignment function. Thanks to its robustness toward illumination variations, occlusions and multi modality, it has been widely used in medical applications for alignment as well as in general tracking problems. Despite those previous works, no highlight has been given on the problem of Hessian computation that yields, in the case of common approximations, to divergence of the optimization process. In this paper we focus on the need of computing the second order derivative of the mutual information in visual servoing. Experiments on a 6 dof robot demonstrates the significance of this work on visual servoing tasks.**

I. Introduction

Visual servoing consists in using the information provided by a vision sensor to control the movements of a dynamic system [1]. Most of the proposed approaches requires the extraction of a set of geometric visual features that have to be tracked and matched over frames. This process has proved to be a difficult one.

Recently, it has been shown that no other information than the image intensity can be considered to control the robot motion and that these difficult tracking and matching processes can be totally removed. The approaches proposed by [2]–[4], [7], no longer require any matching or tracking process. They turn the visual servoing problem into a non linear optimization problem [9]. The error to be minimized is no more defined by the difference between some desired and current features but by an alignment function between the current image $\mathbf{I}$ and the image acquired at the desired position $\mathbf{I}^*$. In [2] the alignment function is defined by the sum of squared differences on the intensity of all pixels of the two images, in [7] by spatial sampling kernels and in [3] by the mutual information between the two images [12]. Whereas these methods are very different from the classical geometric approaches [1], the goal remains the same: from its current pose $\mathbf{r}$, the robot has to reach the desired pose $\mathbf{r}^*$. In terms of optimization, it means that during all the visual servoing task the pose of the robot has to evolve in the direction of the extremum of the choosed alignment function by sending a velocity to the robot. This robot velocity is computed using the derivatives of the cost function with respect to the pose $\mathbf{r}$.

In the present work we focus on the cost function defined by mutual information [10] shared by the current and desired image [3]. This function does not compare directly intensities of the two images but the distribution of the information in the images. This yields to very interesting properties for visual servoing: this approach is robust to large light variations, to occlusions and to the mutlimodality of acquisition between the desired and the current image.

We consider this cost function derivatives and show that special care has to be taken in the computation of the Hessian matrix. Our studies reveal that classical approximation on the Hessian [5], [6], [11] may involve divergence in the visual servoing task.In alignment or tracking algorithm this problem can be solve of using an approach like Brent's method but in visual servoing it is impossible to use this kind of method because it requires backtracking. In previous works [3] the problem has been handled by successively changing some parameters in the computation during the servoing task. This leads to problematic non continuous and non smooth control law and non optimal robot 3D trajectory. To the best of our knowledge, despite the fact that mutual information is commonly used in computer vision, no paper raises the problem of these regularly used approximations.

This paper clearly defines the previous encountered problem and proposes a way to properly handle it. The exact formulation of the Hessian matrix is defined and several experiments on a 6 dof robot show that the behaviour in the 3D space is better than in our previous works. The multimodal image-based navigation proposed in previous work that was limited to 3 dof is now workable with 6 dof applications as it is shown on the last experiment.

The remainder of this paper is organized as follows. The first section presents the way to achieve visual servoing by a non linear optimization and introduces an example that allows illustrating the main expressions. The definition of mutual information is given in Section II with its derivatives where we focus on the Hessian computation. In Section III a solution to perform the visual servoing task despite the small concavity domain of mutual information is proposed. Finally Section IV illustrates the results obtained using a 6 dof robot.

II. Background

A. Visual servoing as a non linear optimization

The aim of a visual servoing task is to minimize the error between the current pose of the robot and a desired pose using a camera. Here we consider an eye-in-hand configuration, the camera is placed at the end effector of the robot, and the scene is supposed motionless. Hypotheses are simple, only the image at the desired pose is known. Mutual information is a function that defines the quantity of information shared by two variables [12]. Maximizing mutual information between the desired image and the current image acquired by the

Amaury Dame is with CNRS, IRISA, Rennes, France. Eric Marchand is with Université de Rennes 1, INRIA lagadic team, France. This work is supported by DGA under contribution to student grant. firstname.name@irisa.fr.

robot is equivalent to minimize the error between the current and the desired position [3]. The optimization process is a Newton's descent method that typically requires the gradient $\mathbf{G}$ and the Hessian matrix $\mathbf{H}$ of the function to maximize. Within this context, the velocity $\mathbf{v}$ sent to the robot can simply be defined by:

$$\mathbf{v} = -\lambda \mathbf{H}^{-1}\, \mathbf{G}^{\top} \tag{1}$$

where λ is the step size of the optimization. Following sections give definitions of mutual information and its derivatives (gradient and Hessian matrix).

B. Illustration

The goal of the visual servoing task is to control a 6 dof robot. Since the possibilities to represent the expressions of a 6 dof non linear optimization problem are limited, a set of illustrations is shown on a more simple example that uses a one degree of freedom visual servoing task. This example allows illustrating the main expressions that we will define in this work. The camera is positioned around the desired pose along the degree of freedom and the results used in the minimization can be analysed. Figure 1 illustrates the corresponding method: an image is acquired at the desired pose of the robot, then the robot goes through surrounding poses using the choosed transformation (here the translation along the x axis of the camera frame is represented) and mutual information and its derivatives are computed using the current image and the desired image for each position.

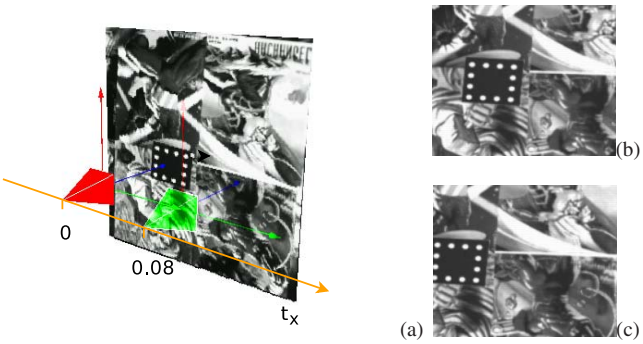


Fig. 1. Method used to compute mutual information and its derivatives along one axis. Here the represented transformation is the translation along the x axis of the camera. (a) External view of the camera at the desired position (red) and at the position corresponding to a 4cm translation (green). (b) image $\mathbf{I}^*$ acquired at the desired pose (c) image $\mathbf{I}$ at the current position.

III. Mutual Information in Visual Servoing

A. Mutual information definition

As shown in previous works [3], mutual information between the desired image $\mathbf{I}^*$ and the current image $\mathbf{I}$ can be defined with respect to the camera pose $\mathbf{r}$ by the following expression:

$$MI(\mathbf{r}) = \sum_{i,j} p_{ij}(i,j,\mathbf{r}) \log\left(\frac{p_{ij}(i,j,\mathbf{r})}{p_i(i,\mathbf{r})p_j(j)}\right) \tag{2}$$

where $p_{ij}(i,j)$ is the probability of the couple $(\overline{\mathbf{I}}(\mathbf{x}), \overline{\mathbf{I}^*}(\mathbf{x}))$ to have the value (i,j) that is called joint probability. p_i and p_j are the probabilities of respectively $\overline{\mathbf{I}}(\mathbf{x})$ and $\overline{\mathbf{I}^*}(\mathbf{x})$ to have the values i and j, these are called marginal probabilities. $\overline{\mathbf{I}}$ and $\overline{\mathbf{I}^*}$ are respectively the images $\mathbf{I}$ and $\mathbf{I}^*$ scaled to belong to the $[0; Nc] \subset \mathbb{R}$ space:

$$\overline{\mathbf{I}}(\mathbf{x}) = \mathbf{I}(\mathbf{x})\frac{N_c}{N_{c_\mathbf{I}}} \qquad \overline{\mathbf{I}^*}(\mathbf{x}) = \mathbf{I}^*(\mathbf{x})\frac{N_c}{N_{c_{\mathbf{I}^*}}}. \tag{3}$$

where $N_{c_\mathbf{I}}$ and $N_{c_{\mathbf{I}^*}}$ are the maximal intensities of the pixels of $\mathbf{I}$ and $\mathbf{I}^*$ (typically 255). The probabilities are basically defined using normalized histogram functions, so that the joint probability is obtained using the joint histogram of the two images and the marginal probabilities are obtained using the histogram of each images as follows:

$$p_{ij}(i,j,\mathbf{r}) = \frac{1}{N_\mathbf{x}} \sum_\mathbf{x} \phi\left(i - \overline{\mathbf{I}}(\mathbf{x},\mathbf{r})\right) \phi\left(j - \overline{\mathbf{I}^*}(\mathbf{x})\right) \tag{4}$$

$$p_i(i,\mathbf{r}) = \frac{1}{N_\mathbf{x}} \sum_\mathbf{x} \phi\left(i - \overline{\mathbf{I}}(\mathbf{x},\mathbf{r})\right) \tag{5}$$

$$p_j(j) = \frac{1}{N_\mathbf{x}} \sum_\mathbf{x} \phi\left(j - \overline{\mathbf{I}^*}(\mathbf{x})\right) \tag{6}$$

where $N_\mathbf{x}$ is the number of point $\mathbf{x}$ in the region of interest in the image. Typically the ϕ function used to compute histograms are Gaussians centered on zero. It is commun to approximate Gaussians with B-splines functions. In [3], ϕ used to be approximated with a first order B-spline that is easy and fast to compute but that is not differentiable and causes difficulties to compute the derivatives needed for the minimization. In our experiments Gaussians is then approximated by tricubic B-splines functions that is a two-times differentiable function (see figure 2).

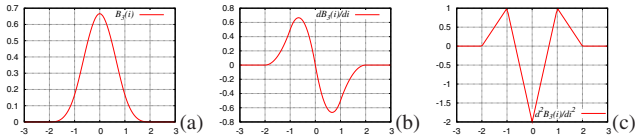


Fig. 2. ϕ function defined as a third order B-spline that is two-times differentiable. (a) third order B-spline, (b) its derivative and (c) its second order derivative.

Mutual information has been computed using the previous definition for the example described in Section II.B. Figure 3(a) shows that mutual information reaches the maximum value when the robot is at the desired pose (corresponding to a null translation along the x axis).

B. Gradient

The gradient of mutual information is its derivative with respect to the camera pose $\mathbf{r}$. As explained in [11], applying chain rules on the general definition of equation (2) and simplifying, the final expression of the gradient can be written:

$$\mathbf{G} = \frac{\partial MI}{\partial \mathbf{r}}(\mathbf{r}) = \sum_{i,j} \frac{\partial p_{ij}}{\partial \mathbf{r}} \left(1 + \log\left(\frac{p_{ij}}{p_i}\right)\right). \tag{7}$$

To compute this gradient the first derivative of the joint probability with respect to the position $\frac{\partial p_{ij}}{\partial \mathbf{r}}$ is needed. Using the previous definition of (4) and since ϕ is two-times

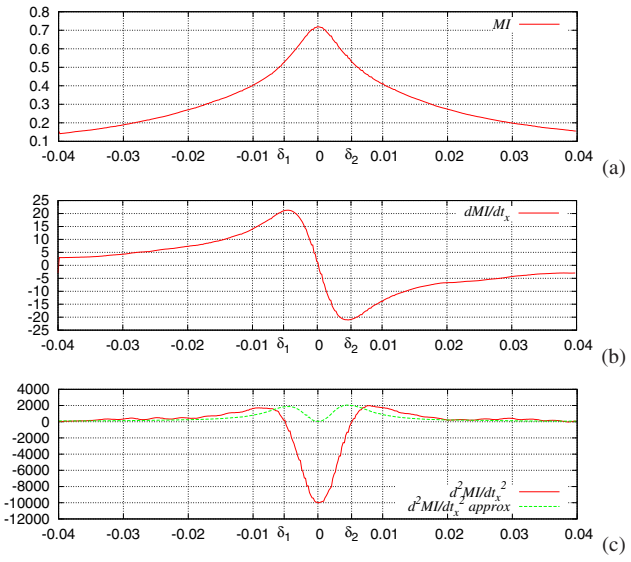


Fig. 3. Computation of mutual information and its derivatives with respect to the horizontal translation of the camera frame in meters. (a) Mutual information, (b) first derivative of mutual information and (c) second order derivatives with and without approximations. $[\delta_1, \delta_2]$ represents the concavity domain of mutual information.

differentiable, the derivative of the joint probability is given by:

$$\frac{\partial p_{ij}}{\partial \mathbf{r}} = \frac{1}{N_{\mathbf{x}}} \sum_{\mathbf{x}} \frac{\partial \phi}{\partial \mathbf{r}} \left(i - \bar{\mathbf{I}}(\mathbf{x}, \mathbf{r})\right) \phi\left(j - \overline{\mathbf{I}^*}(\mathbf{x})\right). \quad (8)$$

Finally the derivative of the ϕ function is given in [3] by:

$$\frac{\partial \phi(i - \bar{\mathbf{I}}(\mathbf{x}, \mathbf{r}))}{\partial \mathbf{r}} = -\frac{\partial \phi(i - \bar{\mathbf{I}}(\mathbf{x}, \mathbf{r}))}{\partial i} \nabla \bar{\mathbf{I}}\, \mathbf{L}_{\mathbf{x}} \quad (9)$$

where $\nabla \bar{\mathbf{I}}$ is the image gradient in the metric space $\left(\nabla \bar{\mathbf{I}}_x, \nabla \bar{\mathbf{I}}_y\right)$ and $\mathbf{L}_{\mathbf{x}}$ is the interaction matrix that links the displacement of a point with the velocity of the robot. To illustrate the results with respect to a single translation along the x axis with a perspective projection, $\mathbf{L}_{\mathbf{x}}$ is set to $[-X/Z \ 0]^\top$ for each point of coordinates (X, Y, Z) in the camera frame.

In the case of visual servoing experiments using the 6 dof robot, the interaction matrix is defined as in [1] using:

$$\mathbf{L}_{\mathbf{x}} = \begin{bmatrix} -1/Z & 0 & x/Z & xy & -(1+x^2) & y \\ 0 & -1/Z & y/Z & 1+y^2 & -xy & -x \end{bmatrix}$$

where x and y are theimage point coordinates. Figure 3(b) shows the computed values of the derivative. These values are consistent with the corresponding mutual information values (Figure 3(a)).

C. Hessian

The Hessian of mutual information is its second order derivative with respect to the camera pose $\mathbf{r}$. Differentiating the previously obtained gradient given by (7) it yields to:

$$\mathbf{H} = \frac{\partial \mathbf{G}}{\partial \mathbf{r}} \quad (10)$$
$$= \sum_{i,j} \frac{\partial p_{ij}}{\partial \mathbf{r}}^\top \frac{\partial p_{ij}}{\partial \mathbf{r}} \left(\frac{1}{p_{ij}} - \frac{1}{p_i}\right) + \frac{\partial^2 p_{ij}}{\partial \mathbf{r}^2} \left(1 + \log \frac{p_{ij}}{p_i}\right)$$

It is classical to consider the second term of this expression as null [6], [11] that gives the following approximation:

$$\mathbf{H} \simeq \sum_{i,j} \frac{\partial p_{ij}}{\partial \mathbf{r}}^\top \frac{\partial p_{ij}}{\partial \mathbf{r}} \left(\frac{1}{p_{ij}} - \frac{1}{p_i}\right). \quad (11)$$

We will see that this approximation is too coarse. In fact, at convergence it yields to a null Hessian. Using a classical non-linear optimization such as a Newton's method (see equation (1)) with this approximation leads to a divergence at the desired position since the inverse of the null Hessian is used to compute the velocity. The complete expression of the Hessian is finally given and, using some examples, the results of the approximation and the exact method will be compared.

1) Approximation results: At convergence the current image is supposed to be similar to the desired image. In the case of an ideal positioning at $\mathbf{r}^*$ we have $\mathbf{I} = \mathbf{I}^*$ which implies that:

$$P\left[\bar{\mathbf{I}}(\mathbf{x}, \mathbf{r}^*) \neq \overline{\mathbf{I}^*}(\mathbf{x})\right] = 0 \quad (12)$$
$$P\left[\bar{\mathbf{I}}(\mathbf{x}, \mathbf{r}^*) = i \cap \overline{\mathbf{I}^*}(\mathbf{x}) = i\right] = P\left[\bar{\mathbf{I}}(\mathbf{x}, \mathbf{r}^*) = i\right] \quad (13)$$
$$= P\left[\overline{\mathbf{I}^*}(\mathbf{x}) = i\right] \quad (14)$$

where $P(\chi)$ is the probability of the event χ. The joint probability is then the diagonalization of the marginal probability $p_j(j)$ with $p_{ij}(i, j, \mathbf{r}) = p_j(j)$ for $i = j$ and 0 otherwise. As a consequence the summation of equation (11) is null for $i = j$.

Moreover considering equation (8), it is clear that the derivative of the joint probability for $\bar{\mathbf{I}} = \overline{\mathbf{I}^*}$ and for $i \neq j$ is quasi null. At the desired pose the all summation of (11) can finally be considered as null.

As expected using the example explained in Section II we can see in Figure 3(c) that the approximated Hessian is null at the desired position $t_x = 0$.

2) Exact Hessian computation: The second part of the expression (10) that has been considered as null in previous works [3] following [6], [11] has then to be taken into account. This involves computing the Hessian of the joint probability p_{ij} with respect to $\mathbf{r}$:

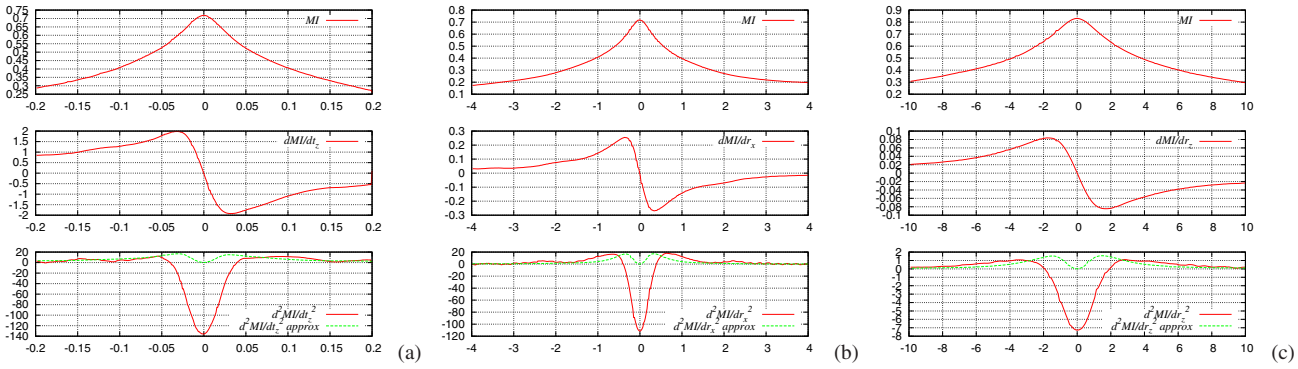
$$\frac{\partial^2 p_{ij}}{\partial \mathbf{r}^2}(i, j, \mathbf{r}) = \sum_{\mathbf{x}} \frac{\partial^2 \phi}{\partial \mathbf{r}^2} \left(i - \bar{\mathbf{I}}(\mathbf{x}, \mathbf{r})\right) \phi\left(j - \overline{\mathbf{I}^*}(\mathbf{x})\right). \quad (15)$$

Using the first derivative of ϕ in (9) and applying chain rules and product derivative, the following result is obtained for the second order derivative:

$$\frac{\partial^2 \phi}{\partial \mathbf{r}^2} \left(i - \bar{\mathbf{I}}(\mathbf{x}, \mathbf{r})\right) = \frac{\partial^2 \phi}{\partial i^2} \left(i - \bar{\mathbf{I}}(\mathbf{x}, \mathbf{r})\right) (\nabla \bar{\mathbf{I}}\, \mathbf{L}_{\mathbf{x}})^\top (\nabla \bar{\mathbf{I}}\, \mathbf{L}_{\mathbf{x}})$$
$$- \frac{\partial \phi}{\partial i} \left(i - \bar{\mathbf{I}}(\mathbf{x}, \mathbf{r})\right) \left(\nabla \bar{\mathbf{I}}_x \mathbf{H}_{\mathbf{x}} + \nabla \bar{\mathbf{I}}_y \mathbf{H}_{\mathbf{y}}\right)$$
$$- \frac{\partial \phi}{\partial i} \left(i - \bar{\mathbf{I}}(\mathbf{x}, \mathbf{r})\right) \mathbf{L}_{\mathbf{x}}^\top \nabla^2 \bar{\mathbf{I}}\, \mathbf{L}_{\mathbf{x}} \quad (16)$$

where $\nabla^2 \bar{\mathbf{I}} \in \mathbb{R}^{2\times 2}$ is the gradient of $\nabla \bar{\mathbf{I}}$ in the metric space and $\mathbf{H}_{\mathbf{x}}$ and $\mathbf{H}_{\mathbf{y}}$ are respectively the derivatives of the first and second line of the interaction matrix $\mathbf{L}_{\mathbf{x}}$ (see [8] for the computation of the two Hessian matrices).

We previously chose the ϕ function as a two-times differentiable function so that the computation of the previous



(a) (b) (c)

Fig. 4. Mutual information and its derivatives with respect to different degrees of freedom: (a) translation along the z axis in meters, (b) rotation around the x axis and (c) rotation around the z axis in degrees.

expression is possible. The Hessian has been computed considering the one degree of freedom example. The figure 3(c) shows the corresponding results and that the final values are accurate.

To validate the previous computation on the 6 dof problem, the same experiments as in figure 3 have been realized on the other degrees of freedom. Since we are working in the camera frame, translations along the x and y axis can be considered as similar, as well as rotations around the x and y axis. So, to illustrate our results in 6 dof, we will only represent the x and z translations and the x and z rotations. Figure 4 represents the corresponding results. The same conclusion as the previous example can be made, the computed Hessian values remain consistent with the value of the gradient.

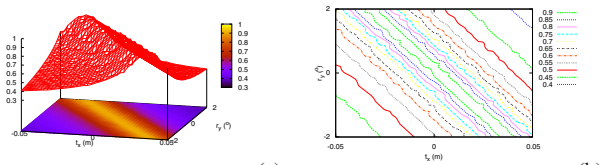
IV. Maximization problem

The analysis in one dimension of the mutual information and its derivatives highlights one issue: the domain of concavity around the desired pose is small. In the case of our first example (see figure 3) the domain of concavity $[\delta_1, \delta_2]$ corresponds to a 1cm translation for a scene at a distance of one meter. Using the classical non-linear optimization as Newton's method minimization defined in (1) will only allow an initial pose belonging to concavity domain.

Nevertheless the shape of mutual information cost function suggests that using a gradient descent method would practically leads to a convergence domain larger than a 8cm translation. Nevertheless if we consider the maximization by simply using the rotation around the y axis and the translation along the x axis (see the corresponding mutual information in figure 5), mutual information has a shape of valley near the convergence. In the case of such valley it is known that gradient descent is unadapted and induces oscillating effects.

An usual method in visual servoing [1] is to replace the current Hessian by the Hessian at the desired position $\mathbf{H}^*$. In our case $\mathbf{H}^*$ is obtained computing the Hessian of mutual information considering $\overline{\mathbf{I}} = \overline{\mathbf{I}^*}$. That is the Hessian of a locally concave function. In this case the entire domain is considered as concave. Then an optimization such as Newton's method theoretically converges to the desired position.

The limit of this method is that mutual information has been chosen for its robustness toward multiple variations. In the case of large variations, the current Hessian at the desired pose differs from the Hessian computed using $\overline{\mathbf{I}} = \overline{\mathbf{I}^*}$, leading to possible problem of convergence.



(a) (b)

Fig. 5. Valley shape of mutual information on the 2 dof (t_x, r_y) subspace: (a) shape of the cost function and (b) isocontours. t_x and r_y are producing a quasi similar transformation to the image acquired by the image.

To overcome this issue the proposed method consist of using the Hessian at the desired position until the robot reaches the concavity domain. At this moment the control law changes and uses the current Hessian. To do so a detection process is created to detect the entrance in the concavity domain: both the theoretical velocities using the current Hessian $\mathbf{v}_c$ and the ones at the desired position $\mathbf{v}_d$ are computed. If the robot is in the convexity domain then the current Hessian is the one of a local convex function whereas the other Hessian is the one of a local concave function, leading to two completely different velocities. As soon as the robot reaches the concavity domain, the two computed velocities are becoming similar. A function $\mu(\mathbf{v}_c, \mathbf{v}_d)$ is simply defined to measure the similarity of the two velocities:

$$\mu(\mathbf{v}_c, \mathbf{v}_d) = \frac{1}{\|\mathbf{v}_c\|^2 \|\mathbf{v}_d\|^2} \mathbf{v}_c^\top \mathbf{v}_d \tag{17}$$

if μ is equal to one the two velocities are aligned. If μ is superior to a given threshold then the robot is considered to be in the concavity domain and the velocity $\mathbf{v}_c$ computed with the current exact Hessian is finally used.

V. Experimental results

A. Standard visual servoing task

To demonstrate the impact of the Hessian computation, experiments have been realized using the proposed method on a 6 dof Gantry robot equipped with a camera mounted on its end-effector. The experimental scheme is the same used in [3]. The robot is moved at the desired pose $\mathbf{r}^*$ to acquire the desired image $\mathbf{I}^*$. Then the robot is moved to a random initial pose. The velocity computed using the control law

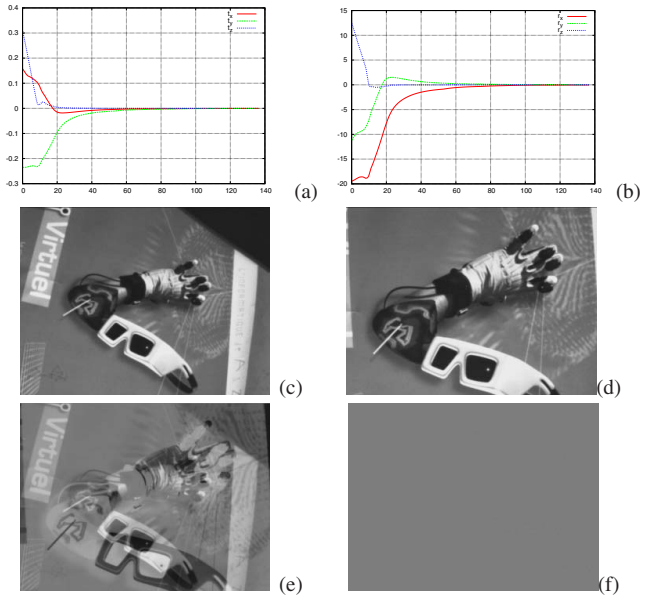


Fig. 6. Experiment of mutual information visual servoing using second order derivatives computation. (a) translation part of $\Delta\mathbf{r}$ (meter) and (b) rotational part of $\Delta\mathbf{r}$ () with x axis in seconds. (c) Initial image, (d) desired image, (e) initial images difference and (f) final images difference $\overline{\mathbf{I}}^* - \overline{\mathbf{I}}$.

defined in (1) is send to the robot. To analyse the behaviour of this method, current camera pose $\mathbf{r}$ and the transformation $\Delta\mathbf{r}$ between $\mathbf{r}^*$ and $\mathbf{r}$ are stored during the task.

Figure 6 shows the acquired images and the corresponding behaviour of the robot for a typical experiment. The initial error pose $\Delta\mathbf{r} = (-15\text{cm}, -23\text{cm}, 30\text{cm}, -19°, -11°, 13°)$ is large. Figure 7 shows the trajectory of the camera in the 3D space and its orientation. This experiment shows that the behaviour of the robot is far more better than in our previous works [3]. There is no more non continuity in the control law that was due to the changes in the parameter Nc (see figure 4 in [3]), here the bin size Nc is fixed to $Nc = 8$. This leads to a smooth trajectory. Contrary to previous works the trajectory of the camera in the 3D space is really satisfying: the first part of the trajectory is almost a straight line and when the camera reaches the valley of the cost function (see figure 5) then the camera is reaching the final pose using a circular trajectory focusing on the scene.

The accuracy of the proposed method remains very high. The robot reaches a pose error $\Delta\mathbf{r}$ below 0.1mm in translation on each axis and $0.01°$ in rotation despite a distance to the scene of approximately 1 meter.

The computation of the exact value of the Hessian could be seen as a time-consuming task. However using a research type code, the time of computation remains reasonable. Using a $2.6GHz$ processor a new velocity is computed each $30ms$ for a frame of size 320×240.

B. Multimodal image-based navigation

The previous experiment gives qualitative results on the method and can be seen as difficult to compare with previous works. To validate the improvments of the proposed method, the navigation experiment presented in [3] that was limited to 3 dof is studied again with 6 dof.

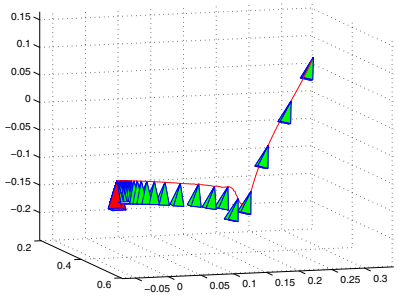


Fig. 7. 3D representation of the trajectory of the camera. Axis are in meters. The desired camera pose is the red pose.

In this experiment we consider the navigation task defined in the sensor space by a database of images acquired during a learning step. This experiment evaluates the robustness of mutual information toward multimodality. So during the learning step, the camera is moving along the desired trajectory and the set of desired images $\mathbf{I}^*(t)$ is acquired on a map scene. In the navigation task the map scene is replaced by a satellite scene. These map and aerial image have been acquired using the *IGN (Institut Géographique Nationale) geoportail* (http://www.geoportail.fr) which is a tool similar to google earth. Map and aerial images have the same scale 1:25000.

We suppose that the first camera pose is near the first desired image $\mathbf{I}^*(t_0)$. The visual servoing task is then used to reach the next intermediate desired pose. The switch between two desired images is performed when the gradient of the mutual information is below a given threshold. The same process is used until the desired image is the last image of the set of images $\mathbf{I}^*(t)$.

In previous work only the three dof (t_x, t_y, r_z) were considered since the approximation in the non-linear optimization uses to make it impossible to deal with the valley of the cost function induced by the two couples of transformation (t_x, r_y) and (t_y, r_x) (see figure 5).

In the experiment the six camera dof have to be considered to track the considered trajectory. Figure 9 shows the desired trajectory of the camera on the scene and the resulting one. Figure 8 shows the corresponding images acquired during the experiment. As we can see on both figures the middle part of the desired trajectory is defined as a (t_x, r_y) transformation: the second and third columns of figure 8 look the same whereas the transformation between the corresponding positions is a $13cm$ translation along x with a $0.11°$ rotation around y.

Since the real transformation between the map and satellite scene are not precisely known, it is not possible to compare numerically the result with a ground truth. Then, results are evaluated on the 3D resulting trajectory of the camera and on the superposition of the desired and current images. We do not consider image error since the difference of acquisition modalities makes such error insane.

Figure 8 shows that the two images are correctly aligned and figure 9 shows that the two trajectories are similar. We can conclude that the navigation task gives accurate results.

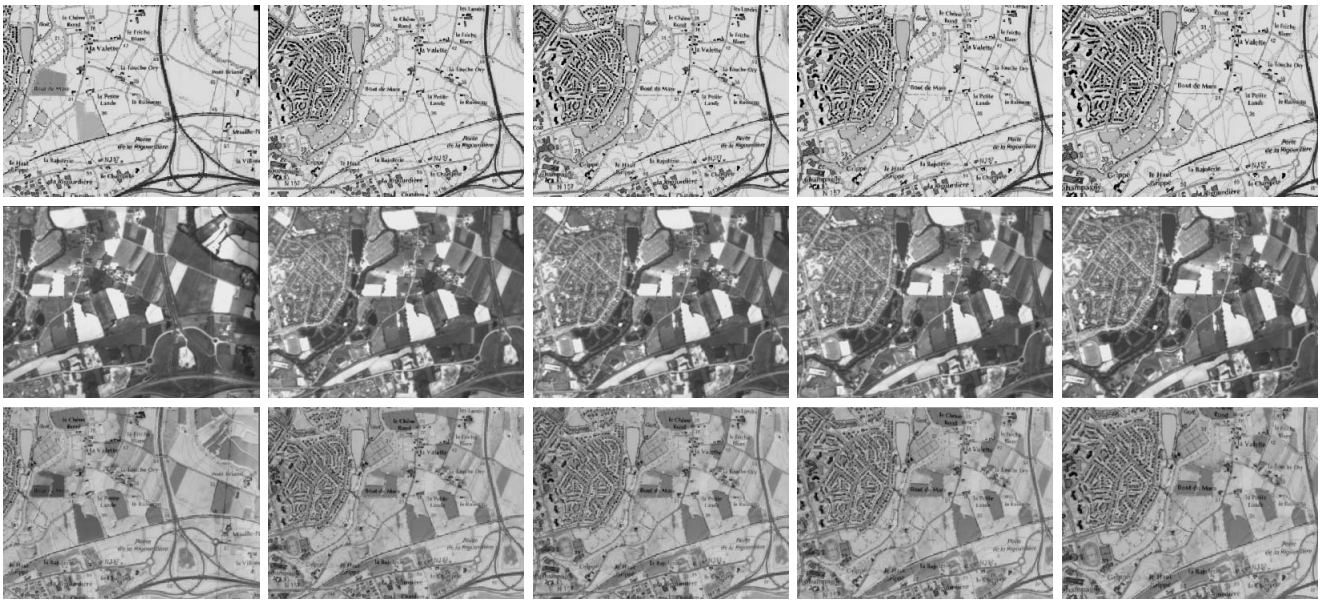
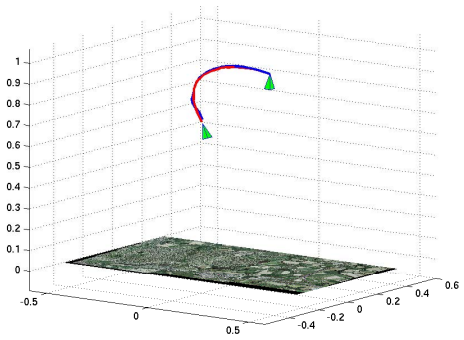


Fig. 8. Multi-modal visual servoing in a navigation task. First row: desired images (acquired during the learning step) ; second row: current images ; third row : desired images overlaid on the current ones.



(a)

Fig. 9. Reference path of the camera in red and resulting path in blue on the scene in multimodal navigation experiment with initial and final position of the camera. Axis units are in meters.

It demonstrates the robustness of the proposed method compared to the previous one even with a complex 6 dof trajectory tracking task.

VI. Conclusion

In this paper we focused on mutual information-based visual servoing. The new proposed scheme shows interesting properties since it is robust to illumination variations, occlusions and different modality of acquisition between the reference and the current image. Furthermore it does not require any feature extraction or matching/tracking process.

Nevertheless the initial approach [3] used a common approximation that makes it depend on parameter adjustment during the visual servoing task. This adjustment caused the trajectory of the camera not to be smooth. The issue led by the approximation is explained in this paper and solutions have been proposed to overcome it. A particular care is taken on the second order derivation of mutual information in information theory-based visual servoing. This solution proves its benefits in different applications showing a better behaviour on a 6 dof robot.

This study could be generalized to tracking algorithms using mutual information since tracking is similar to virtual visual servoing. Moreover the method proposed for the non-linear optimization that yields to a good trajectory of the camera in the 3D space could give similar results in SSD based visual servoing.

References

[1] F. Chaumette and S. Hutchinson. Visual servoing and visual tracking. In B. Siciliano and O. Khatib, editors, *Handbook of Robotics*, chapter 24, pages 563–583. Springer, 2008.

[2] C. Collewet, E. Marchand, and F. Chaumette. Visual servoing set free from image processing. In *IEEE Int. Conf. on Robotics and Automation, ICRA'08*, Pasadena, CA, May 2008.

[3] A. Dame and E. Marchand. Entropy based visual servoing. In *IEEE Int. Conf. on Robotics and Automation, ICRA'09*, pages 707–713, Kobe, Japan, May 2009.

[4] K. Deguchi. A direct interpretation of dynamic images with camera and object motions for vision guided robot control. *Int. Journal of Computer Vision*, 37(1):7–20, June 2000.

[5] N. Dowson and R. Bowden. Mutual information for lucas-kanade tracking (milk): An inverse compositional formulation. In *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 30:180–185, January 2008.

[6] N.D.H. Dowson and R. Bowden. A unifying framework for mutual information methods for use in non-linear optimisation. In *ECCV'06*, volume 1, pages 365–378, June 2006.

[7] V. Kallem, M. Dewan, J.P. Swensen, G.D. Hager, and N.J. Cowan. Kernel-based visual servoing. In *IEEE/RSJ Int. Conf. on Intelligent Robots and System, IROS'07*, pages 1975–1980, San Diego, USA, October 2007.

[8] J.T. Laprest and Y. Mezouar. A Hessian approach to visual servoing. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS'04*, Sendai, Japan, 2004.

[9] E. Malis. Improving vision-based control using efficient second-order minimization techniques. In *IEEE Int. Conf. on Robotics and Automation, ICRA'04*, volume 2, pages 1843–1848, New Orleans, April 2004.

[10] C. E. Shannon. A mathematical theory of communication. *Bell system technical journal*, 27, 1948.

[11] P. Thvenaz and M. Unser. Optimization of Mutual Information for Multiresolution Image Registration. *IEEE Transactions on Image Processing*, 9(12):2083–2099, 2000.

[12] P. Viola and W. Wells. Alignment by maximization of mutual information. *Int. Journal of Computer Vision*, 24(2):137–154, 1997.