Direct visual servoing in the frequency domain

Eric Marchand

***Abstract*—In this paper, we propose an original approach to extend direct visual servoing to the frequency domain. Whereas most of visual servoing approaches relied on the geometric features, recent works have highlighted the importance of taking into account the photometric information of the entire images. This leads to direct visual servoing (DVS) approaches. In this paper we propose no longer to consider the image itself in the spatial domain but its transformation in the frequency domain. The idea is to consider the Discrete Cosine Transform (DCT) which allows to represent the image in the frequency domain in terms of a sum of cosine functions that oscillate at various frequencies. This leads to a new set of coordinates in a new precomputed orthogonal basis, the coefficients of the DCT. We propose to use these coefficients as the visual features that are then considered in a visual servoing control law. We then exhibit the analytical formulation of the interaction matrix related to these coefficients. Experimental results validate our approach.**

***Index Terms*—Visual servoing, Sensor-based control**

I. Introduction

VISUAL servoing uses the information provided by a vision sensor to control the movements of a dynamic system [5]. This approach requires the extraction of visual information (usually geometric features) from the image in order to design the control law.

While there has been progress in extracting and tracking relevant features, a new approach called direct visual servoing (DVS) has been emerging for almost 10 years now [14], [7], [6], [9], [8]. It has been demonstrated that only the pixel intensities of the images can be taken into account to control the robot's motion and that conventional tracking and matching processes can be avoided. Nevertheless, it feature a small convergence domain compared to classical techniques. Various schemes have been proposed in order to improve the robustness of DVS by considering various descriptors (image intensity, gradient, color, etc.) or cost functions (mutual information [9], histogram distances [3], mixture of Gaussians [8]. Another solution to increase the convergence domain would be to extract from the image a set of coefficients that could then be used as control input. The idea is not to extract geometric features from the image but to "compress" the original image information in order to get a compact representation (dimensionality reduction problem). This is what has been done with photometric moments which allows the preservation of geometric information [2]. It was shown that it provides a better behavior than a classical control based on points [5] and extends significantly the convergence of the photometric visual servoing approach [6]. In [19], [10], [16] the image is projected on a new basis thanks to a Principal Component Analysis (PCA) process (also known as Karhunen-Loève expansion). The control is then performed on the image coordinates in the eigenspace (an orthogonal basis). This process requires the off-line computation of this eigenspace and then, for each new frame, the projection of the image on this subspace in order to compute the set of coordinates (coefficients) in the new basis that will be used in the control law. With respect to [19], [10] where the interaction matrix is estimated on-line, [16] exhibit an explicit and analytical formulation of the interaction matrix. Recently, it has been proposed to consider convolutional neural network to bypass the modelling step [4].

In this paper we propose no longer to consider the image itself (the spatial domain) but its transformation in the frequency domain. The idea is to consider the Discrete Cosine Transform (DCT [1]) which allows to represent the image in the frequency domain in terms of a sum of cosine functions that oscillate at various frequencies: the coefficients of the DCT. The coefficients with large amplitude (high energy) are associated with the lower frequencies of the image. The DCT is very useful for image compression (eg, in the JPEG standard [21]) since it has a strong "energy compaction" property meaning that most of the image information is concentrated in a few low-frequency components [1]. Our goal is then to transform, thanks to the DCT, the image from the spatial to the frequency domain and then use the coefficients of the DCT to build a new control law. Our contributions are:

- we show how the coefficients of the DCT can be considered within a visual servoing control law;
- we exhibit an explicit and analytical formulation of the related interaction matrix;
- we propose a method to drastically reduce the dimensionality of the problem by considering only the low frequencies. This allows to reduce the noise and have a smoother cost function thus improving the performance and enhance the convergence area;
- we show on various experiments including real 6 DoF positioning tasks, that these approaches allow large displacements and a satisfactory decrease of the error norm thanks to a well modelled interaction matrix.

Few visual servoing schemes consider the frequency domain. Let us note however that the magnitude of the Fourrier transform has been considered to control translation and rotation along and around the optical axis [14] and Fourier shift property has been used to estimated 2D translation (from which a classical control law is built) in [18]. More closely related to our approach [20], [11] consider the coefficients of wavelets or shearlets which is a transform in the time-frequency domain. Nevertheless, dimensionality reduction was not sought (although, as in our case, a wise selection of the

Manuscript received: September 9th, 2019; Revised December 8th, 2019; Accepted January 1st, 2020.This paper was recommended for publication by Editor Cesar Cadena upon evaluation of the Associate Editor and Reviewers' comments.

Eric Marchand is with Univ Rennes, Inria, CNRS, IRISA, Rennes, France Eric.Marchand@irisa.fr.

Digital Object Identifier (DOI): see top of this page.

wavelets coefficient should improve the control law behavior). As far as this later point is concerned, this approach is closely related to PCA-based visual servoing [16]. It achieves similar results although with the proposed new approach, the transform that allows a change in coordinates is not learnt but precomputed.

In the reminder of this paper, Section II gives an overview of the DVS scheme. Section III gives an overview of the Discrete Cosine Transform. Section IV gives the details of the control laws including the derivation of the related interaction matrix. Finally, Section V illustrates the effectiveness of the approach with experiments carried out in simulation and on a 6 DoF robot.

II. Direct Visual Servoing

A. Positioning task by visual servoing

The aim of a positioning task is to reach a desired pose of the camera $\mathbf{r}^*$, starting from an arbitrary initial pose. To achieve that goal, one needs to define a cost function that reflects, in the image space, this error. Most of the time this cost function is an error measure which needs to be minimized. Considering the actual pose of the camera $\mathbf{r}$ the problem can therefore be written as an optimization process:

$$\widehat{\mathbf{r}} = \arg\min_{\mathbf{r}} \mathbf{e}(\mathbf{r}). \tag{1}$$

which consists in minimizing an error $\mathbf{e}(\mathbf{r}) = \mathbf{s}(\mathbf{r}) - \mathbf{s}^*$, usually expressed in the image space, between what the camera sees (a set of features $\mathbf{s}(\mathbf{r})$) and what it wants to see (i.e., the desired configuration of the visual features $\mathbf{s}^*$).

This visual servoing task is achieved by iteratively applying a velocity to the camera. This requires the knowledge of the interaction matrix $\mathbf{L_s}$ related to **s(r)** that links the variation of $\dot{\mathbf{s}}$ to the camera velocity $\mathbf{v}$ and which is defined as [12], [5]:

$$\dot{\mathbf{s}}(\mathbf{r}) = \mathbf{L_s}\mathbf{v}. \tag{2}$$

The control law is classically given by [5]:

$$\mathbf{v} = -\lambda \mathbf{L_s^+} \mathbf{e}(\mathbf{r}) \tag{3}$$

where λ is a positive scalar and $\mathbf{L_s^+}$ is the pseudo inverse of $\mathbf{L_s}$.

B. Photometric visual servoing

Recent works propose to directly use the information provided by the entire image [7], [6]. In [6], a control law was proposed that minimizes the error between the current image and the desired one. In that case the vector of visual features in nothing but the image itself and the error to be regulated is the sum of squared differences (the SSD).

In that case, the feature $\mathbf{s}$ becomes the image itself ($\mathbf{s}(\mathbf{r}) = \text{vec}(\mathbf{I}(\mathbf{r}))$). $\text{vec}(\mathbf{I})$ denotes the vectorization [13] of image matrix $\mathbf{I}$. This means that the optimization process becomes [6]:

$$\widehat{\mathbf{r}} = \arg\min_{\mathbf{r}} (\text{vec}(\mathbf{I}(\mathbf{r})) - \text{vec}(\mathbf{I}^*)) \tag{4}$$

where $\mathbf{I}(\mathbf{r})$ and $\mathbf{I}^*$ are respectively the image seen at the position $\mathbf{r}$ and the template image (both of N^2 pixels assuming squared $N \times N$ images). The control law (Photo-VS) is given by:

$$\mathbf{v} = -\lambda \mathbf{L_I^+} (\text{vec}(\mathbf{I}(\mathbf{r})) - \text{vec}(\mathbf{I}^*)) \tag{5}$$

where λ is a positive scalars and $\mathbf{L_I}$ is the interaction matrix related to the luminance [15]. If we introduce the interaction matrices $\mathbf{L}_x$ and $\mathbf{L}_y$ related to the coordinates x and y of a pixel $\mathbf{x}$, the 1×6 interaction matrix $\mathbf{L_{I(x)}}$ for each pixel $\mathbf{x}$ is given by [15], [6]:

$$\mathbf{L_{I(x)}} = -(\nabla I_x \mathbf{L}_x + \nabla I_y \mathbf{L}_y) \tag{6}$$

where ∇I_x and ∇I_y are the components along x and y of the image gradient ∇I. The complete $N^2 \times 6$ interaction matrix $\mathbf{L_I}$ is obtained by stacking the $\mathbf{L_{I(x)}}$ for all the pixel $\mathbf{x}$ of the image $\mathbf{I}$. This approach features many advantages: first, it does not require any matching or tracking process; second, since the image measurements are nothing but the pixel intensity, there are no error in the feature extraction process leading to a very precise realization of the task. Nevertheless, the main drawback of DVS is its small convergence domain compared to classical techniques, which is due to the high non-linearities of the cost function to be minimized.

III. Background on the DCT

The DCT allows to transform an image from the spatial domain to the frequency domain. It uses the property that the intensity of two neighbor pixels are usually highly correlated. The transformation (change of coordinates) attempts to decorrelate the image data. A few coefficients contain most of the information (corresponding to the image low frequencies).

A. The Discrete Cosine Transform

The discrete cosine transform [1] is a linear function that expresses a 1D signal $I(x)$ in the frequency domain in terms of a sum of cosine functions that oscillate at various frequencies. There are variant of the DCT, let us consider here one of the most common: the DCT-II. For a Signal $I(x)$ the coefficient are given by:

$$f(u) = \frac{1}{\sqrt{2}} \sum_{x=0}^{N-1} I(x) \cos\big(\frac{u(2x+1)\pi}{2N}\big), u = 0..N-1 \tag{7}$$

Multidimensional variants exist and if an image $\mathbf{I}(x, y)$ (that is nothing but a 2D signal) whose size in $N \times N$ is considered, we have:

$$\mathbf{F}(u,v) = \alpha \sum_{y=0}^{N-1} \sum_{x=0}^{N-1} \mathbf{I}(x,y) \cos\big(\frac{u(2x+1)\pi}{2N}\big) \cos\big(\frac{v(2y+1)\pi}{2N}\big) \tag{8}$$

with

$$\alpha = \frac{2}{N} C(u) C(v) \tag{9}$$

where $C(u)$ and $C(v)$ are normalization constants given by:

$$C(u) = 1/\sqrt{2} \text{ if } u = 0, \ 1 \text{ otherwise} \tag{10}$$

The magnitude of the coefficients $\mathbf{F}(u, v)$ drops very rapidly as u and v increases (see Figure 1a-b).

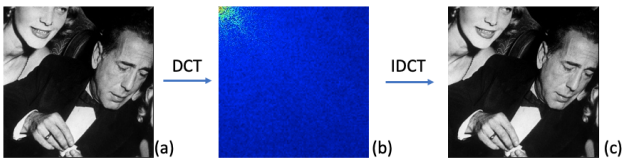

Figure 1. DCT computation and IDCT: (a) original image $\mathbf{I}(x, y)$ (b) Coefficients $\mathbf{F}(u, v)$ after the DCT (c) Image reconstruction using the IDCT.

B. The DCT matrix transform

For a 1D DCT, one can see that, from equation (7), $f(u) = \mathbf{C}\, I(x)$ where $\mathbf{C}$ is a matrix defined by:

$$C_{ij} = \begin{cases} 1/\sqrt{N} & \text{if } i = 0 \\ \sqrt{2/N} \cos\left(\frac{i(2j+1)\pi}{2N}\right) & \text{otherwise} \end{cases} \quad (11)$$

Note that the columns of $\mathbf{C}$ form an orthogonal basis ($\mathbf{C}$ is thus orthogonal). $\mathbf{C}\,\mathbf{I}$ is an $N \times N$ matrix whose columns contain the one-dimensional DCT of the columns of $\mathbf{I}$. The two-dimensional DCT of $\mathbf{I}$ can be computed as:

$$\mathbf{F} = \mathbf{C}\,\mathbf{I}\,\mathbf{C}^\top \quad (12)$$

which is equivalent to equation (8). Since $\mathbf{C}$ is orthogonal, and thus that $\mathbf{C}^{-1} = \mathbf{C}^\top$, equation (12) is nothing but a change of coordinates in a new basis. This computation is faster than using (8) because $\mathbf{C}$ needs to be computed only once. We will also see that this formulation is very suitable for our visual servoing purpose.

C. Inverse DCT

From equation (12), it is easy to see that the DCT is invertible leading to the IDCT (Inverse DCT, see Figure1c):

$$\mathbf{I} = \mathbf{C}^\top\,\mathbf{F}\,\mathbf{C} \quad (13)$$

Indeed, from equation (12), multiplying by $\mathbf{C}^\top$ from the left and then by $\mathbf{C}$ on the right, we have $\mathbf{C}^\top\mathbf{F}\mathbf{C} = \mathbf{C}^\top\mathbf{C}\mathbf{I}\mathbf{C}^\top\mathbf{C} = \mathbf{I}$ considering that $\mathbf{C}$ is orthogonal. We will see in the experiments section that the use of IDCT is useful for visualization since we can recover the original image (up to some artifacts if all the coefficients in $\mathbf{F}$ are not considered, see Figure 2c).

IV. DCT AND VISUAL SERVOING

We now present how the DCT can be used within a visual servoing control law.

A. DCT coefficients as visual features and their interaction matrix

The main idea is to consider the coefficients of the DCT as the visual features in our visual servoing problem. A basic solution to our problem would be to consider all the N^2 coefficients as the visual features. When the camera pose is $\mathbf{r}$, the DCT matrix $\mathbf{F}(\mathbf{r})$ related to image $\mathbf{I}(\mathbf{r})$ is given by

$$\mathbf{F}(\mathbf{r}) = \mathbf{C}\,\mathbf{I}(\mathbf{r})\,\mathbf{C}^\top \quad (14)$$

and the vector $\mathbf{f}(\mathbf{r})$ of visual feature will be then defined by:

$$\mathbf{f}(\mathbf{r}) = \text{vec}(\mathbf{F}(\mathbf{r})) \quad (15)$$

The error to be minimized is then given by:

$$\mathbf{e}(\mathbf{r}) = \mathbf{f}(\mathbf{r}) - \mathbf{f}^* \quad (16)$$

where $\mathbf{f}^*$ is obtained from equation (14) and (15) for the desired image $\mathbf{I}^*$. We will see in the next paragraph, that most of the coefficients within $\mathbf{F}(\mathbf{r})$ are almost null and that a feature selection process can be considered.

Having defined the cost function to be minimized, one has to compute the interaction matrix $\mathbf{L_f} = \frac{\partial \mathbf{f}(\mathbf{r})}{\partial \mathbf{r}}$ that links the variation of $\mathbf{f}(\mathbf{r})$ to the camera motion. $\mathbf{L_f}$ can be efficiently computed by deriving equation (14) for each element of vector $\mathbf{r}$. Let us denote r_i the i-th component ($i = 1..6$) of the pose $\mathbf{r}$. Since r_i is a scalar, it is easy to note that:

$$\frac{\partial \mathbf{F}(\mathbf{r})}{\partial r_i} = \frac{\partial \mathbf{C}\,\mathbf{I}(\mathbf{r})\,\mathbf{C}^\top}{\partial r_i} = \mathbf{C}\,\frac{\partial \mathbf{I}(\mathbf{r})}{\partial r_i}\mathbf{C}^\top \quad (17)$$

where $\frac{\partial \mathbf{I}(\mathbf{r})}{\partial r_i}$ is a $N \times N$ matrix that, in practice, contains the i-th column of matrix $\mathbf{L_I}$: $\frac{\partial \mathbf{I}(\mathbf{r})}{\partial r_i} = \text{vec}^{-1}(\mathbf{L}_{\mathbf{I}\bullet i})$. The interaction matrix $\mathbf{L_f}$ can the be built as:

$$\mathbf{L_f} = \left(\text{vec}\left(\frac{\partial \mathbf{F}(\mathbf{r})}{\partial r_1}\right) \ \ldots \ \text{vec}\left(\frac{\partial \mathbf{F}(\mathbf{r})}{\partial r_6}\right)\right) \quad (18)$$

leading to the desired $N^2 \times 6$ interaction matrix.

B. Control law

The complete control law (DCT-VS) is then given by:

$$\mathbf{v} = -\lambda \mathbf{L_f}^+ \left(\mathbf{f}(\mathbf{r}) - \mathbf{f}^*\right). \quad (19)$$

From a practical point of view we considered a Levenberg-Marquardt-like control law given by:

$$\mathbf{v} = -\lambda \left(\mathbf{H} + \mu\, diag(\mathbf{H})\right)^{-1} \mathbf{L_f}^\top \left(\mathbf{f}(\mathbf{r}) - \mathbf{f}^*\right). \quad (20)$$

with $\mathbf{H} = \mathbf{L_f}^\top \mathbf{L_f}$ is an approximation of the Hessian. More precisely, each component of the gradient is scaled according to the diagonal of the Hessian, which leads to larger displacements along the direction where the gradient is low. Such a control law has proven its effectiveness in a context of DVS [6], [9], [16]. Note that, as in [16], beside gains λ and μ in equation (20) and, obviously K, no parameters are involved in these experiment. In all the experiments described below, we set $\lambda = 1$ and $\mu = 0.01$. μ decreases by a factor 0.99 at each iteration. Therefore, the control law tends to the classical visual servoing control law that is similar to a Gauss-Newton minimization process (see equation (19)).

C. A formal comparison between Photo-VS and DCT-VS control laws

In fact, if all the coefficient of the DCT are considered, it can be proved that, despite the fact that the cost functions and the Jacobians are different, the pure photometric control law (Photo-VS) and the DCT-based control law (DCT-VS) are equivalent.

Indeed, considering that $\mathbf{Y} = \mathbf{A}\,\mathbf{X}\,\mathbf{B}$ is equivalent to $\text{vec}(\mathbf{Y}) = (\mathbf{B}^\top \otimes \mathbf{A})\text{vec}(\mathbf{X})$ (where $\otimes$ denotes the Kronecker product), equation (14) can be rewritten as [13]:

$$\mathbf{f}(\mathbf{r}) = \text{vec}(\mathbf{F}(\mathbf{r})) = (\mathbf{C} \otimes \mathbf{C})\text{vec}(\mathbf{I}(\mathbf{r})) \quad (21)$$

since $\mathbf{C} \otimes \mathbf{C}$ is a constant this is leading to:

$$\mathbf{L_f} = (\mathbf{C} \otimes \mathbf{C})\mathbf{L_I} \tag{22}$$

$\mathbf{L_f}$ and $\mathbf{L_I}$ are $N^2 \times 6$ matrices. Equations (18) and (22) are equivalent, but, unfortunately, $\mathbf{C} \otimes \mathbf{C}$ is a $N^2 \times N^2$ matrix which makes the computation of equation (22) prohibitive in practice with respect to (18). Using this formulation of the interaction matrix, the DCT-VS control law (19) is given by:

$$\mathbf{v} \quad = \quad -\lambda \left((\mathbf{C} \otimes \mathbf{C})\mathbf{L_I} \right)^{+} \mathrm{vec}(\mathbf{C}\mathbf{I}(\mathbf{r})\mathbf{C}^{\top} - \mathbf{C}\mathbf{I}^{*}\mathbf{C}^{\top})$$

Considering that for any orthogonal matrix $\mathbf{C}$, it can be proved that:

$$\left((\mathbf{C} \otimes \mathbf{C})\mathbf{L_I} \right)^{+} \mathrm{vec}(\mathbf{C}\mathbf{I}(\mathbf{r})\mathbf{C}^{\top} - \mathbf{C}\mathbf{I}^{*}\mathbf{C}^{\top}) = \mathbf{L_I}^{+} \mathrm{vec}(\mathbf{I}(\mathbf{r}) - \mathbf{I}^{*})$$

demonstrating that the two control laws (19) and (5) are equivalent.

Working in the frequency domain is thus equivalent as working in the spatial domain. This appears to be a disappointing result. Nevertheless, let us point out that this result is valid only if all the coefficients of the DCT are considered in the vector of visual features $\mathbf{f}(\mathbf{r})$. The main interest of considering the frequency domain is that only a few (well selected) coefficients can be considered. In that case, the DCT transform is no longer bijective and this equivalence is no longer valid. This coefficients selection process is the purpose of the next paragraph.

D. Dimensionality reduction and coefficients selection

Working in the spatial domain, it is obvious that two neighbour pixels (but a few) are highly correlated in term of intensity (there is a high covariance). Keeping all the pixels is then redundant (but a selection process followed by a matching process would be a tedious task). We just demonstrated that working in the spatial domain or in the frequency domain while considering all the frequencies is equivalent. Nevertheless, the main advantage of working in the frequency domain, is that considering all the coefficients of the DCT are not necessary and selecting the optimal frequencies (features) is quite simple.

The original image can be seen as a linear combination of these coefficients with cosine basis functions. Thus, as stated in the section II, the DCT has achieved a change of coordinates and sorted the coefficients in increasing order of frequency. One can easily see on Figure 2a that the coefficients with high amplitude are associated with the lower frequencies (upper left of the DCT image). Let us recall that low frequencies correspond to slow varying information (continuous surface) whereas high frequencies correspond to quickly varying information (edges).

We thus propose to consider K coefficients corresponding to the lower frequencies of the image (see Figure 2b). For a direct visual servoing problem, this has many advantages: first discarding high frequencies (low pass filtering) allows to suppress the noise in the image; second, this also allows to have a smoother cost function thus improving the convergence of the control law. Furthermore, considering only the low frequencies allows to increase the overlapping between large scale structures which also increases the convergence area. This can be seen on Figure 3. When considering all the 40000 coefficients (-in that case DCT-VS (Figures 3b) is equivalent to Photo-VS (Figures 3a)), the cost function features local minima for large displacements. As can be seen on Figures 3a and 3b), it also features a narrow minimum at the middle of a slope plateau with low gradient (leading to a prohibitive number of iterations to reach convergence). Nevertheless, as soon as we reduce the dimensionality of the problem (from $K = 40000$ to $K = 50$), one can see on Figure 3c that the cost function is smoother with a larger convergence area and higher gradient allowing a faster convergence of the control law.

Figure 2. Coefficients selection (a) Coefficients $\mathbf{F}(u, v)$ after the DCT (see Fig 1) (b) Low-pass filtering, Selection of the coefficients using the zig-zag algorithm (c) Image reconstruction using the IDCT.

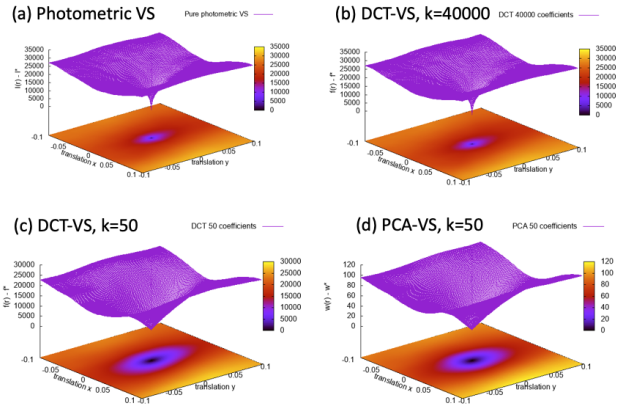

Figure 3. Cost function of various DVS methods (a) pure photometric method [7], (b) DCT-VS with $K = 40000$, (c) DCT-VS with $K = 50$, (d) PCA-VS [16] with $K = 50$. Planes below correspond to the projection of the cost functions (color is a linear mapping of the cost function value).

The coefficients are thus ordered according to the zig-zag sequence [21] (see Figure 4). This ordering places low frequency coefficients before high frequency coefficients which are likely to be zero. When the camera pose is $\mathbf{r}$, the vector $\mathbf{f}(\mathbf{r})$ of visual feature is then the coefficients of the DCT of $\mathbf{I}(\mathbf{r})$ according the zig-zag pattern (see Figure 4). Considering only the K first coefficients along the zig-zag path (ie, $\mathbf{f}(\mathbf{r})$ is a size K vector) is equivalent to achieve a low pass filtering on the image (the IDCT using only few coefficients is shown on Figure 2c). The interaction matrix $\mathbf{L_f}$ (a $K \times 6$ matrix) is built as in equation (18) but it is restricted to the same K coefficients obtained following the very same zig-zag pattern.

A comparison with the PCA-based VS. In [16], for the same reasons, we also proposed to reduce the dimensionality of the image. We proposed to project the image on an orthogonal basis using a principal component analysis (PCA) approach.

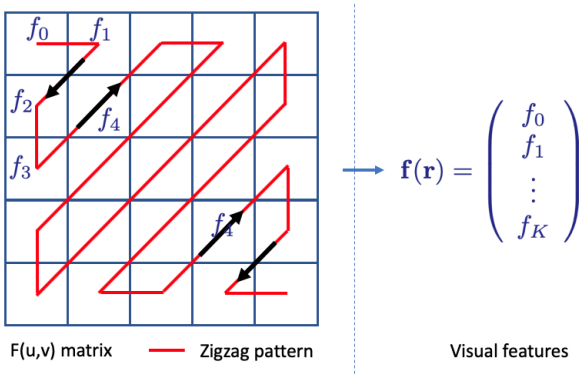

Figure 4. DCT coefficients extraction techniques from the $\mathbf{F}$ matrix: the zigzag method and the definition of our visual feature $\mathbf{f}(\mathbf{r})$.

The PCA is a linear transform where the basis functions are taken from the statistical properties of the image data, and can thus be adaptive. In fact, this basis has to be learnt for each considered scene (which is a tedious task). Nevertheless, it is optimal in the sense of energy compaction. Indeed, it places as much energy as possible in as few coefficients as possible. An interest of such approach is that, when projecting an incoming image on this basis, the greatest variance comes to lie on the first coefficient, the second greatest variance lies on the second coefficient, etc. A control law (PCA-VS) based on this coefficient was also proposed. The control laws PCA-VS and DCT-VS are very similar, only the orthogonal basis is modified. Wrt. PCA-VS, DCT-VS features many advantages but mainly, it is faster to compute and does not require any learning step since the basis is precomputed; thus it is scene agnostic which is an important advantage of the new proposed method. Furthermore, DCT performs closely wrt. the PCA in term of energy compaction [1]. A similar number of coefficients can be considered in both approaches (that is typically $K = 50$). As can be expected the shape of the cost functions and the convergence areas for DCT-VS and PCA-VS are very close (see second row of Figure 3cd). leading to a close behavior as will be seen in the next section.

V. Experimental results

Experiments have been carried out in simulation and on a 6-DOF anthropomorphic robotic arm (a Viper 850 from Adept Company) equipped with a camera mounted on the end-effector. The camera calibration as well as the hand-eye calibration have been done in an off-line step. The image processing and the control law computation are performed on a PC equipped with a 8-cores 3.7 Ghz Intel Xeon. The code has been written in C++ using the ViSP library [17]. The time required for an iteration of the VS closed loop is constant whatever the number of considered coefficients K. In our experiments, an iteration corresponds to $60ms$ (including image acquisition, DCT, interaction matrix and control law computations). Images size are 220×220.

Finally, let us point out that all the reported experiments feature a positioning task. The desired image $\mathbf{I}^*$ (and then $\mathbf{f}^*$) is computed. The robot is moved toward the initial position. The control law intends then to minimize the error $\mathbf{e} = \mathbf{f}(\mathbf{r}) - \mathbf{f}*$. Six degrees of freedom (DoF) are considered in all the experiments.

Figure 5. Experimental setup: camera mounted on a Viper 850 from ADEPT

A. Simulation results

Simulations have first been carried out in order to validate the proposed control laws while allowing a fair comparison of different direct visual servoing approaches (Photo-VS [6], PCA-VS [16]). The error between the initial and desired pose is, in all the cases, $\mathbf{\Delta r} = -0.11m, -0.31m, -0.01m, -25.00, 5.00, 25.00)$[1]. This is a very large initial error for a direct VS scheme. The initial and desired images are shown in Figure 6a and 6b. The initial image reconstructed with the IDCT are shown on Figure 6c and 6d for $K = 20$ and K=50 respectively. The reconstructed image using the PCA is shown on Figure 6e. With small value of K it can be seen that only the low frequencies of the image remain.

Figure 6. Simulated experiment: (a) initial image, (b) desired image, IDCT of the initial image for (c) $K = 20$ (d) $K = 50$, (e) reconstructed image with the PCA with $K = 50$

In the first experiment (Figure 7), we report both the photometric visual servoing approach (Photo-VS) and the

[1]The following notations has been used: $\mathbf{\Delta r} = (\mathbf{t}, \theta\mathbf{u})$, where $\mathbf{t}$ describes the translation part of the homogeneous matrix related to the transformation from the current to the desired frame, while its rotation part is expressed under the form $\theta\mathbf{u}$, where $\mathbf{u}$ represents the unit rotation-axis vector and θ the rotation angle around this axis. This representation is also considered in the plots reporting the positioning errors.

new DCT-VS approach considering all the 48400 coefficients. Let us recall, that when all the coefficients are considered, the control laws are equivalent. Although the control law allows the camera to converge toward the desired position, the cost function is highly non-linear which leads to large perturbation in the velocity plots and a complex 3D trajectory (see Figure 10, blue trajectory). Figure 8 and 9 show the results of the new DCT-VS approach with respectively $K = 20$ and $K = 50$. With respect to Photo-VS, the velocities are smoother, convergence is much faster (600 iterations vs 1200), and the 3D trajectory closer to the geodesic. This is mainly due to the fact that the cost function is far less non-linear. A comparison with PCA-VS [16] with $K = 50$ is also proposed (see Figure 11), the control law behavior is very similar to the DCT-VS method. In Figure 10, the 3D camera trajectories are plotted for Photo-VS, DCT-VS with various number of coefficients, and PCA-VS. As expected reducing the dimensionality of the problem greatly improves the general behavior of the system.

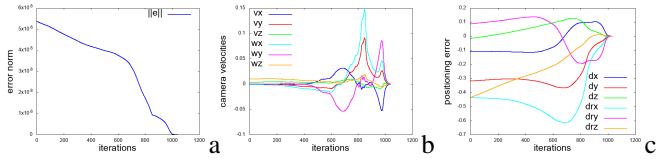

Figure 7. Experiment with Photo-VS or DCT-VS with $K = 48400$. (a) $\| \mathbf{I}(\mathbf{r}) - \mathbf{I}^* \|$ or $\| \mathbf{f}(\mathbf{r}) - \mathbf{f}^* \|$ (b) camera velocity (in m/s and rad/s) (c) positioning error (in m and rad).

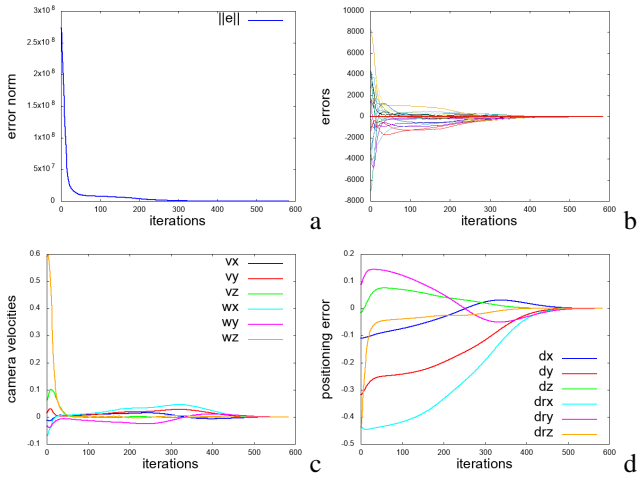

Figure 8. Experiment with DCT-VS with $K = 20$ (a) $\| \mathbf{f}(\mathbf{r}) - \mathbf{f}^* \|$, (b) error $f_i(\mathbf{r}) - f_i^*$, (c) camera velocity (in m/s and rad/s), (c) positioning error (in m and rad).

B. Experimental results on a 6 DoF robot

We first consider a positioning task with respect to a planar scene. The displacement to be achieved is $\mathbf{\Delta r} = (0.04m, 0.27m, 0.04m, 22.3^o, 8^o, 26.3^o)$. The transformation between the initial and desired poses (and particularly the rotation around the x and z axes) is very large and

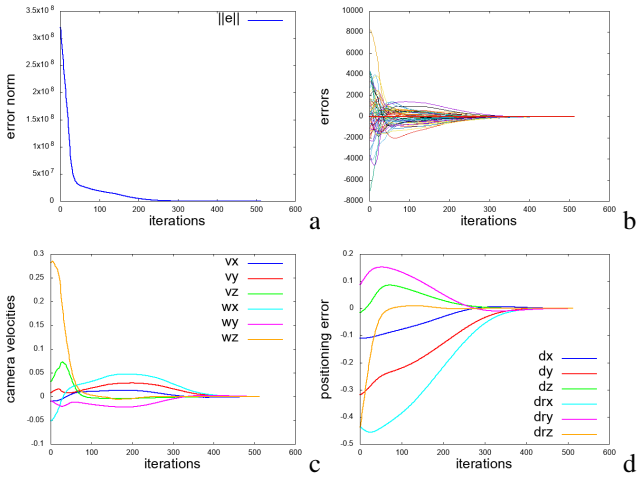

Figure 9. Experiment with DCT-VS with $K = 50$ (a) $\| \mathbf{f}(\mathbf{r}) - \mathbf{f}^* \|$, (b) error $f_i(\mathbf{r}) - f_i^*$, (c) camera velocity (in m/s and rad/s), (c) positioning error (in m and rad).

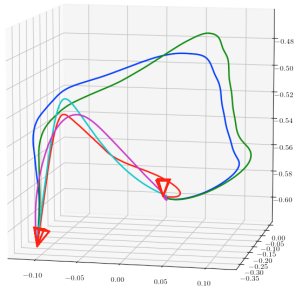

Figure 10. 3D camera trajectories: Photo-VS or DCT-VS with $K = 48400$ in blue, DCT-VS with $K = 10000$ in green, DCT-VS with $K = 50$ in cyan, DCT-VS with $K = 20$ in red, PCA-VS with $K = 50$ in purple.

makes this experiment very challenging. This is also illustrated by the initial and desired images depicted in Figure 13(a-c). We considered only $K = 50$ coefficients. The norm of the cost function $\| \mathbf{f}(\mathbf{r}) - \mathbf{f}^* \|$ decreases monotonously (Figure 13.f). The decrease in errors (Figure 13.g) is also highly satisfactory considering the fact that only the interaction matrix at the desired position and an approximated depth were employed. The final error is $\mathbf{\Delta r} = (0.0016m, 0.0004m, 0.00033m, 0.09^o, -0.11^o, 0.01^o)$ which shows the accuracy of the proposed approach. After iteration 150, μ (equation (20)) gradually decreases leading to a small and temporary augmentation of the cost function (Figure 13f). What can be observed is that the control law mainly achieves a motion along and around the z axes (see Figure 13h, iterations 0-150). Actually, motions along z translation and rotation are the motions that introduce the main difference in the image leading to higher gradients of the cost function. x (resp. y) translation coupled with y (resp. x) rotation are far less observable in the image leading to smaller gradients. Thus, the optimization process tends to drive (mainly but not only) the camera along z translation and rotation axes faster than on the other axes. Then, it compensates for the other axes. In this "second step" the error in the image is quite

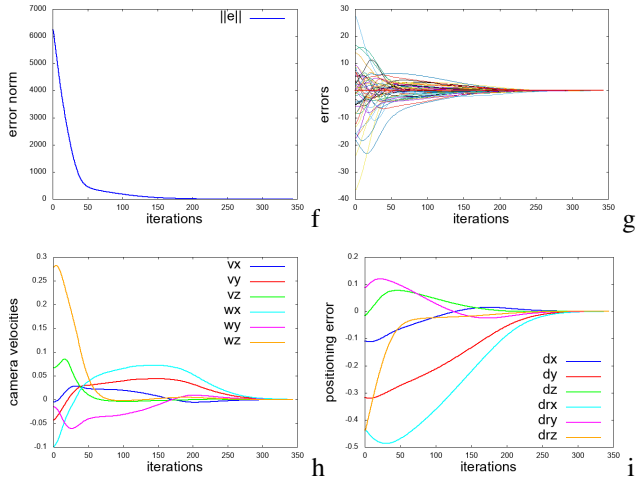

Figure 11. Experiment with PCA-VS [16] with $K = 50$ (a) $\| \mathbf{w}(\mathbf{r}) - \mathbf{w}^* \|$, (b) error $w_i(\mathbf{r}) - w_i^*$, (c) camera velocity (in m/s and rad/s), (c) positioning error (in m and rad).

small although the 3D motion remains large (this can easily be seen comparing Figure 13f and 13i). This pattern can be found in all the reported experiments (see Figure 14). Finaly, note that it is possible to gradually include high-frequency coefficients at later stages of convergence and tend towards a Photo-VS control law. Nevertheless, experiments show that it does not improve significantly the positioning accuracy (indeed, although it adds more details it also adds noise).

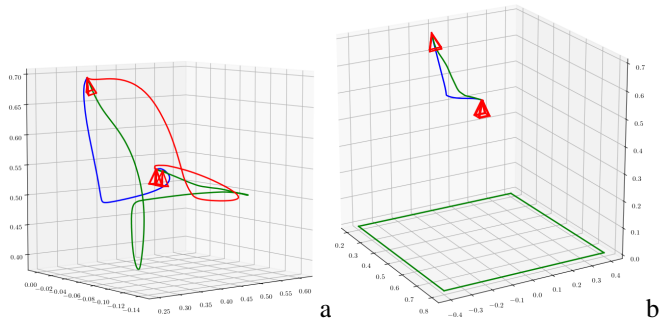

Figure 12. 3D camera trajectories (a) with various number of DCT coefficients (green: $K = 10$; red : $K = 20$; blue : $K = 50$ (b) comparison with PCA-based visual servoing [16] (green: PCA-VS, blue : DCT-VS)

We experimented with various values of K. Figure 12 shows the camera trajectories for $K = 10$ (blue), $K = 20$ (green) and $K = 50$ (red). In all the cases, the visual servoing control law converges although final precision is slightly better with $K = 50$. The camera trajectory is also better with $K = 50$. Considering more coefficients does not improve either the trajectory or the precision.

We also compare our new DCT based visual servoing (DCT-VS) approach with other DVS method: photometric VS [7] and PCA-VS [16]. Photometric VS failed since the motion is too large wrt. to the small convergence area of the method. As far as PCA-VS is concerned, we use the same number of coefficients to reduce the dimensionality of the image: $K = 50$ in both approaches. Although, the trajectory obtained with PCA-VS is slightly closer to the geodesic (see Figure 12b), the precision is better with DCT-VS (in translation: 1.7mm with DCT-VS versus 3.1mm as well as in rotation: 0.14^o versus 0.44^o). Furthermore let us recall that for DCT-VS, no learning step was necessary.

Finally, we also consider various non planar scenes (see Figure 14) with an electronic board, a large electric plug, and a piece of foam with a very repetitiv 3D pattern. The height of the object with respect to the underlying plane is, respectively 5cm, 15cm and 5cm. We consider $K = 50$ for the former and $K = 100$ for the two later since the scenes feature high frequencies patterns (selecting the optimal value of K depending on the scene aspects is a perspective of this work). For all these experiments, the camera converges precisely toward the desired position.

VI. Conclusion

In this paper we demonstrated that direct visual servoing can be achieved in the frequency domain. The image is transformed thanks to the DCT and the control law is built from a few coefficients of the DCT that correspond to the low frequencies of the image. It was also shown that the interaction matrix related to these coefficients can be explicitly and analytically calculated. We also demonstrated that reducing the dimensionality of the problem by adequately selecting the coefficients greatly improves the behavior of the control law. Results show the effectiveness of this approach on various examples.

References

[1] N. Ahmed, T. Natarajan, and K. R. Rao. Discrete cosine transform. *IEEE Transactions on Computers*, C-23(1):90–93, Jan 1974.

[2] M. Bakthavatchalam, O. Tahri, and F. Chaumette. A Direct Dense Visual Servoing Approach using Photometric Moments. *IEEE Trans. on Robotics*, 34(5):1226–1239, October 2018.

[3] Q. Bateux and E. Marchand. Histograms-based visual servoing. *IEEE Robotics and Automation Letters*, 2(1):80–87, January 2017.

[4] Q. Bateux, E. Marchand, J. Leitner, F. Chaumette, and P. Corke. Training deep neural networks for visual servoing. In *IEEE Int. Conf. on Robotics and Automation, ICRA'18*, pages 3307–3314, Brisbane, Australia, May 2018.

[5] F. Chaumette and S. Hutchinson. Visual servo control, Part I: Basic approaches. *IEEE Robotics and Automation Magazine*, 13(4):82–90, December 2006.

[6] C. Collewet and E. Marchand. Photometric visual servoing. *IEEE Trans. on Robotics*, 27(4):828–834, August 2011.

[7] C. Collewet, E. Marchand, and F. Chaumette. Visual servoing set free from image processing. In *IEEE Int. Conf. on Robotics and Automation, ICRA'08*, pages 81–86, Pasadena, CA, May 2008.

[8] N. Crombez, E.M. Mouaddib, and G. Caron. Photometric Gaussian mixtures based visual servoing. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS'15*, pages 5486–5491, Hamburg, Germany, September 2015.

[9] 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.

[10] 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.

[11] L.-A. Duflot, R. Reisenhofer, B. Tamadazte, N. Andreff, and A. Krupa. Wavelet and Shearlet-based Image Representations for Visual Servoing. *The Int. Journal of Robotics Research*, 38(4):422–450, April 2019.

[12] B. Espiau, F. Chaumette, and P. Rives. A new approach to visual servoing in robotics. *IEEE Trans. on Robotics and Automation*, 8(3):313–326, June 1992.

[13] G. Golub and C. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, third edition, 1996.

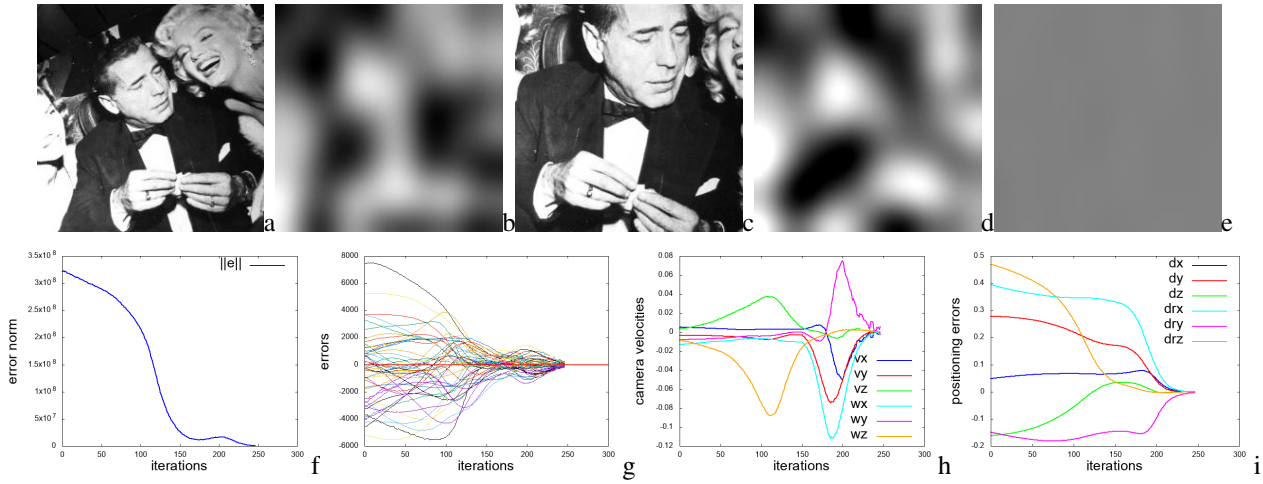

Figure 13. Experiment with real planar scene : we consider $K = 50$ coefficients (a) initial image acquired by the camera $\mathbf{I}(\mathbf{r})$, (b) reconstructed image $IDCT(\mathbf{I}(\mathbf{r}))$ with $K = 50$ acquired from the desired position, (c) desired image $\mathbf{I}^*$ (d,e) error $IDCT(\mathbf{F}(\mathbf{r})) - IDCT(\mathbf{F}^*)$ between reconstructed image for initial and desired position (a,b,c,d,e) are used for visualization but are not used in the algorithm. Only the error $\mathbf{f}(\mathbf{r}) - \mathbf{f}^*$ plotted in (f-g) is considered, (f) $\| \mathbf{f}(\mathbf{r}) - \mathbf{f}^* \|$ (g) $f_i(\mathbf{r}) - f_i^*$ (h) camera velocity (in m/s and rad/s) (i) positioning error (in m and rad).

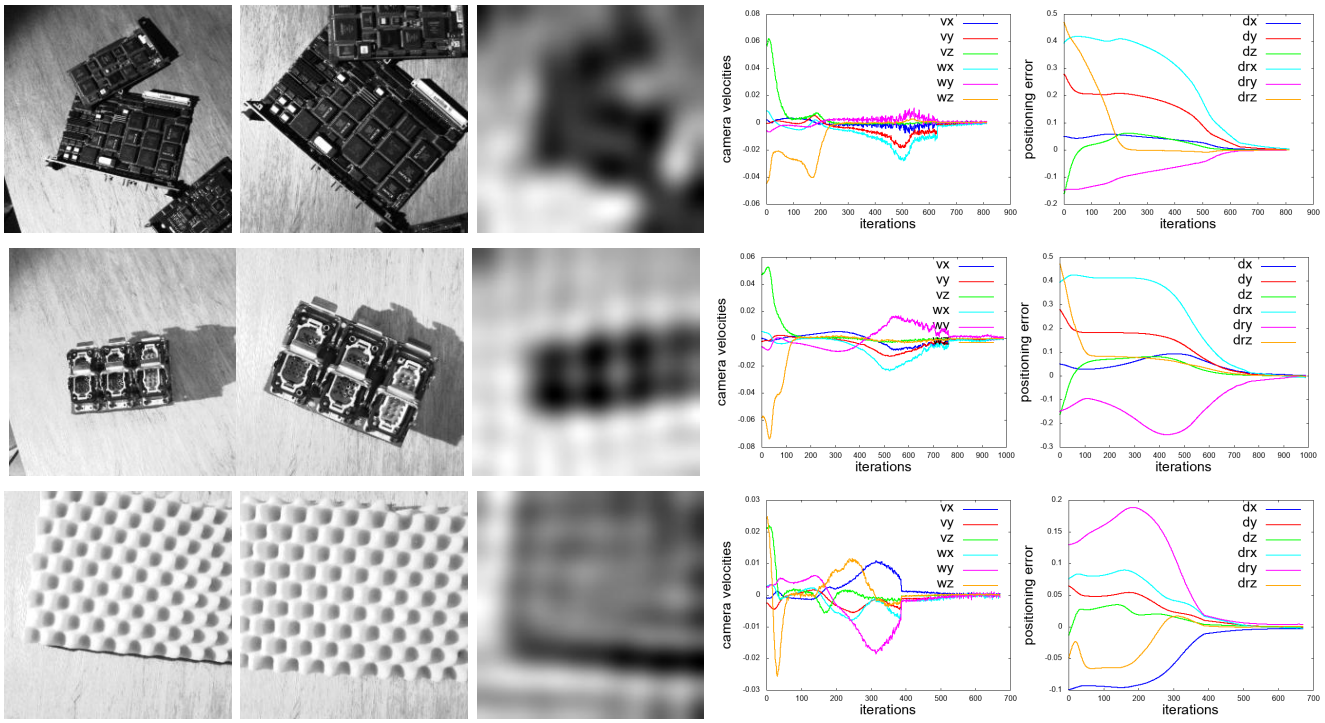

Figure 14. Experiment with real non planar scene (Column 1) initial image acquired by the camera $\mathbf{I}(\mathbf{r})$, (Column 2) desired image $\mathbf{I}^*$, (Column 3) reconstructed image $IDCT(\mathbf{I}(\mathbf{r}))$ (Column 4) camera velocity (in m/s and rad/s) (Column 5) positioning error (in m and rad)

[14] 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.

[15] E. Marchand. Control camera and light source positions using image gradient information. In *IEEE Int. Conf. on Robotics and Automation, ICRA'07*, pages 417–422, Roma, Italia, April 2007.

[16] E. Marchand. Subspace-based visual servoing. *IEEE Robotics and Automation Letters*, 4(3):2699–2706, July 2019.

[17] 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 and Automation Magazine*, 12(4):40–52, December 2005. Special Issue on "Software Packages for Vision-Based Control of Motion", P. Oh, D. Burschka (Eds.).

[18] N. Marturi, B. Tamadazte, S. Dembélé, and N. Piat. Visual servoing schemes for automatic nanopositioning under scanning electron microscope. In *IEEE Int. Conf. on Robotics and Automation, ICRA'14*, pages 981–986, May 2014.

[19] S.K. Nayar, S.A. Nene, and H. Murase. Subspace methods for robot vision. *IEEE Trans. on Robotics*, 12(5):750 – 758, October 1996.

[20] M. Ourak, T. Brahim, O. Lehmann, and N. Andreff. Direct visual servoing using wavelet coefficients. *IEEE/ASME Transactions on Mechatronics*, 24(3):1129–1140, June 2019.

[21] G. K. Wallace. The jpeg still picture compression standard. *Communications of the ACM*, 34(4):30–44, April 1991.