Visual Servoing using Trifocal Tensor

Marwan OSMAN

Contents

1 Introduction

Visual servoing is an approach for controling the motion of a robotic system from visual measurements. Many works have been realized in the past in this area, and is mainly divided into two main categories: Image-based, and Pose-based Visual Servoing. The trifocal tensor is well known in computer vision for tracing geometric information from three images of the same scene. The purpose of this thesis is to design an uncalibrated visual servoing method of a 6-DOF mainpulator or robot based on the three-view projective geometry properties. Few studies where conducted on this work but they didn't provide a generic analytical solutions for 6-DOF robots. This method differs than the two main visual servoing approaches as the control loop is closed over projective measures, which are the trifocal tensor elements. These projective measures are found directly from images across three views, without explicitly recovering the camera pose or directly closing the loop in the image space. The trifocal tensor geometric model is more robust than the two view geometry models as it involves the information given by a third view, and the set of correspondences obtained is more robust to outliers.

2 Trifocal Tensor

2.1 Tensor Notation

Tensors are geometric objects used to represent linear relations between vectors, scalars, and other tensors. A tensor can be represented as a multi-dimensional array of numerical values. The order of a tensor is the dimensionality of the array needed to represent it. Scalars are single numbers and are thus 0th-order tensors. Vectors are 1-dimensional array, 1st-order tensors arranged in a column or row. Matrices are 2-dimensional arrays, 2nd-order tensors arranged as a 2D array of numbers. Similarly, a tensor with three indices may be thought as a 3D array of numbers.

Tensors provide a natural and concise mathematical framework for formulating and solving problems in areas of physics. Tensors express the relationship between vectors, hence they are independent of a particular choice of coordinate system.

The notation for a tensor is similar to that of a matrix, except that a tensor may have an arbitrary number of indices $e.g.$:$A_{ijk\ldots}$. In addition, a tensor with rank $r+s$ may be of mixed type (r,s), consisting of r `contravariant (upper)` indices and s `covariant (lower)` indices. In tensor notation, a vector v would be written v_i, where $i = 1, \cdots, m$, and a matrix is a tensor of type $(1,1)$ would be written as A_i^j.

Tensor notation can provide a very concise way of writing vector and more general identities. For example, the dot product $u.v$ can be simply written as

$$u.v = u_i v^i$$

where repeated indices are summed over. This is called `Einstein Summation`. It is a notational convention for simplifying expressions including summations of vectors, matrices and general tensors. The convention can be best illustrated through the following equation

$$c_k^i = a_j^i b_k^j = \sum_j a_{ij} b_{jk}$$

Similarly, the cross product can be concisely written as

$$(u \times v)_i = \epsilon_{ijk} u^j v^k,$$

where ϵ_{ijk} is the permutation tensor defined for $r, s, t = 1, \cdots, 3$ as follows:

$$\epsilon_{rst} = \begin{cases} 0 & \text{unless } r, s \text{ and } t \text{ are distinct} \\ +1 & \text{if } rst \text{ is an even permutation of 123} \\ -1 & \text{if } rst \text{ is an odd permutation of 123} \end{cases}$$

2.2 Three-View Geometry

The Trifocal Tensor is a $3 \times 3 \times 3$ array of numbers that incorporates all projective geometric relationships among three views. It relates the coordinates of corresponding points or lines in three views, being independent of the scene structure and depending only on the relative motion among the three views and their intrinsic calibration parameters. Hence, the trifocal tensor can be considered as the generalization of the fundamental matrix in three views. It can also be seen as a collection of three rank-two 3×3 matrices T_1, T_2, T_3.

The geometric basis for the trifocal tensor can be deduced from the incidence relationship of three corresponding lines. We start by supposing a line 3-space is imaged in three views as in Figure.1. The planes back-projected from the lines in each view must all meet in a single line in space, the 3D line that projects to the mated line in the three images. Since in general three arbitrary planes in space do not meet in a single line, this geometric incidence condition provides a genuine constraint on sets of corresponding lines.

Let $l_i \leftrightarrow l_i' \leftrightarrow l_i''$ be the set of corresponding lines of L. Let the camera matrices for the three views be $P = [I \mid 0]$, $P' = [A \mid a_4]$, $P'' = [B \mid b_4]$,

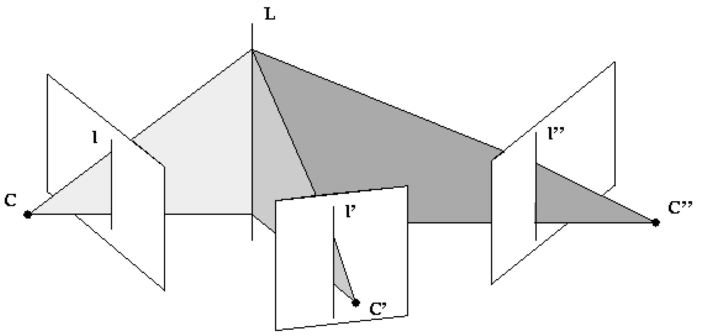

Figure 1: Trifocal geometry of three views

where A and B are 3×3 matrices, and the vectors a_i and b_i are the `i`-th columns of the respective camera matrices for $i = 1, \cdots, 4$. A and B are the homographies from the first to the second and third cameras respectively. Due to our choice for the first camera matrix, a_4 and b_4 are consequently the epipoles of the first camera in views two and three respectively.

$$a_4 = e' = P'C \qquad\qquad b_4 = e'' = P''C$$

Each image line back-projects to a plane as shown in Figure.1. These planes are

$$\pi = P^T l = \begin{pmatrix} l \\ 0 \end{pmatrix} \qquad \pi' = P'^T l' = \begin{pmatrix} A^T l' \\ a_4^T l' \end{pmatrix} \qquad \pi'' = P''^T l'' = \begin{pmatrix} B^T l'' \\ b_4^T l'' \end{pmatrix}$$

The intersection constraint of the three planes in the common line in 3-space can be expressed algebraically by the requirement that the 4×3 matrix $M = \begin{bmatrix} \pi & \pi' & \pi'' \end{bmatrix}$ has rank 2. Points on the line of intersection may be represented as $X = \alpha X_1 + \beta X_2$, with X_1 and X_2 linearly independent. Such points lie on all three planes and so $\pi^T X = \pi'^T X = \pi''^T X = 0$. It follows that $M^T X = 0$. Consequently M has a 2-dimensional null-space since $M^T X_1 = 0$ and $M^T X_2 = 0$.

Since the rank of M is 2, there is a linear dependence between its columns m_i, such that.

$$M = \begin{bmatrix} m_1, m_2, m_3 \end{bmatrix} = \begin{bmatrix} l & A^T l' & B^T l'' \\ 0 & a_4^T l' & b_4^T l'' \end{bmatrix}$$

$$m_1 = \alpha m_2 + \beta m_3$$

From the 2nd row of M, $\alpha = k(b_4^T l'')$ and $\beta = -k(a_4^T l')$ for some scalar k. Applying this back to the 1st row get

$$l = (b_4^T l'') A^T l' - (a_4^T l') B^T l''$$

$$l = (l''^T b_4) A^T l' - (l'^T a_4) B^T l''$$

The `i`-th coordinate l_i of l may be written as

$$l_i = l''^T (b_4 a_i^T) l' - l'^T (a_4 b_i^T) l''$$
$$l_i = l'^T (a_i b_4^T) l'' - l'^T (a_4 b_i^T) l''$$

This relationship can be expressed with the permutation tensor such as

$$\mathcal{T}_i = a_i b_4^T - a_4 b_i^T \tag{1}$$
$$l_i = l'^T \mathcal{T}_i l'' \tag{2}$$

$\mathcal{T}$ is then the trifocal tensor relating the 3 views together. It has 27 elements. There are 26 independent rations apart from the common overall scaling factor of the tensor. However, the tensor has only 18 independent degrees of freedom. Each of 3 camera matrices has 11 degrees of freedom which makes 33 in total. However, 15 degrees of freedom must be subtracted to account for the projective world frame, thus leaving 18 degrees of freedom.

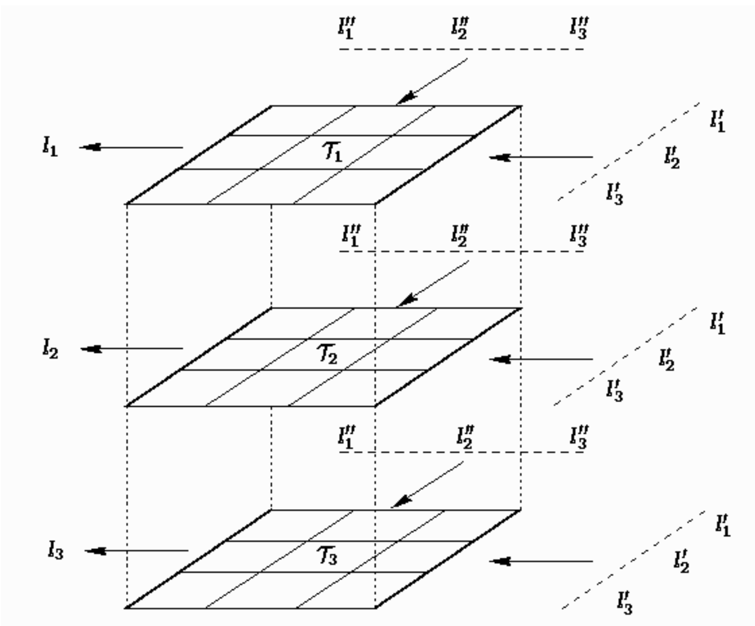

Figure 2: **A 3D representation of the trifocal tensor** $l_i = l'_j l''_k \mathcal{T}_i^{jk}$

A point x on the line l must satisfy $x^T l = \sum_i x^i l_i = 0$. From (2), this may be written as $l'^T (\sum_i x^i \mathcal{T}_i) l'' = 0$ which means that there exists a 3D point X mapping to x in the first image, and to points on the lines l' and l'' in the second and third images. It's the incidence relationship that holds point-line-line correspondence.

To develop the point-point-point correspondence relationship, we can explore the homography map between the points on first and third images given by $x'' = Hx$ and $l = H^T l''$. Thus we get

$$h_i = \mathcal{T}_i^T l'$$

Similarly, the homography from the first to the second views

$$h_i = \mathcal{T}_i l''$$

Using these results back to our point-line-line correspondence relation,

$$x'' = (\sum_i x^i \mathcal{T}_i^T) l' x''^T [x'']_\times = l'^T (\sum_i x^i \mathcal{T}_i)[x'']_\times = 0^T$$

The line l' passes through x' may be written as $l' = [x']_\times y'$ for some point y' on l'. Then

$$y'^T [x']_{times} (\sum_i x^i \mathcal{T}_i)[x'']_\times = 0^T$$

This relation holds true for all lines l' through x' and so is independent of y', hence this relation can be written as

$$[x']_{times} (\sum_i x^i \mathcal{T}_i)[x'']_\times = 0_{3\times 3}$$

which expresses the point-point-point coincidence relationship required. Expressing this relation in proper tensor notation is then given by

$$x^i (x'^j \epsilon_{jpr})(x''^k \epsilon_{kqs}) \mathcal{T}_i^{pq} = 0_{rs} \tag{3}$$

2.3 Recovering Projection Matrices

Since the trifocal tensor embeds the geometry of the three cameras in our scene, this implies that the camera matrices may be computed from the trifocal tensor up to a projective ambiguity [1].

First, the epipoles e', e'' are retrieved. Let u_i and v_i be the left and right null-vectors respectively of $\mathcal{T}_i$, *i.e.:* $u_i^T \mathcal{T}_i = \mathbf{0}^T, \mathcal{T}_i v_i = 0$. Then the epipoles are obtained as the null vectors to the following 3×3 matrices:

$$e'^T [u_1, u_2, u_3] = 0 \text{ and } e''^T [v_1, v_2, v_3] = 0 \tag{4}$$

Next, to retrieve the camera matrices P', P''. The epipoles are normalized to unit norm, then:

$$P' = [[\mathcal{T}_1, \mathcal{T}_2, \mathcal{T}_3] e'' | e'] \text{ and } P'' = [(e'' e''^T - I)[\mathcal{T}_1^T, \mathcal{T}_2^T, \mathcal{T}_3^T] e' | e''] \tag{5}$$

3 Control Law

To formulate the control law of the trifocal tensor based visual servoing, we need to establish the model of the system and find the relation between the input of the control and the trifocal tensor coefficients. The trifocal tensor can be computed from feature correspondences across the three views. Our goal is to find the interaction matrix relating the control input and the tensor derivatives.

$$\dot{\mathcal{T}}_{(jkl)} = L_{\mathcal{T}_{(jkl)}} \begin{pmatrix} v_c \\ w_c \end{pmatrix} \tag{6}$$

3.1 Coefficients of the Trifocal Tensor

To compute the relation between the coefficients of the trifocal tensor and the Projective matrices, first we write the Tensor relation (1) with the proper visual servoing notation as follows

$$\mathcal{T}_{(jkl)} = a_{(kj)}b_{(4l)} - a_{(4k)}b_{(lj)} \tag{7}$$

For the sake of maintaining the Visual Servoing convention, we omit the use of i as an index for the tensor relation. The letter i further on will be indicating the initial view in the control system. The Camera positions are then written as C_{c^*}, C_c, C_i for the desired, current, and initial camera positions respectively. The same may also be done for the projection matrices, and instead of using the matrices A, B to express the homography between the first and second or third views, we will use the standard notation from Visual Servoing with the corresponding Rotation matrices ${}^cR_{c^*}, {}^iR_{c^*}$ respectively. For the translations a_4, b_4, we will use ${}^ct_{c^*}, {}^it_{c^*}$ respectively. Leading superscripts will denote the frame with respect to which a set of coordinates are defined. Thus, the rotation matrix ${}^cR_{c^*}$ gives the orientation of the desired camera frame relative to the current camera frame.

$$\begin{aligned} C &\Rightarrow C_{c^*} \text{ Desired camera position} \\ C' &\Rightarrow C_c \text{ Current camera position} \\ C'' &\Rightarrow C_i \text{ Initial camera position} \end{aligned}$$

$$\begin{aligned} P_d = P &= [I|0] \\ P_c = P' &= [A|a_4] : {}^c\mathbf{M}_{c^*} = [{}^c\mathbf{R}_{c^*}|{}^c\mathbf{t}_{c^*}] \\ P_i = P'' &= [B|b_4] : {}^i\mathbf{M}_{c^*} = [{}^i\mathbf{R}_{c^*}|{}^i\mathbf{t}_{c^*}] \end{aligned}$$

The Tensor relation (7) can then be expressed as follows

$$\mathcal{T}_{(jkl)} = {}^{c}R_{c^*(kj)}\ {}^{i}t_{c^*(l)} - {}^{c}t_{c^*(k)}\ {}^{i}R_{c^*(lj)} \tag{8}$$

Expanding this relation further to compute the tensor coefficients is straight forward, expanding for values of j, k, l we get the following tensor coefficients:

$$\begin{aligned}
\mathcal{T}_{(111)} &= {}^{c}R_{c^*(11)}\ {}^{i}t_{c^*(1)} - {}^{c}t_{c^*(1)}\ {}^{i}R_{c^*(11)} \\
\mathcal{T}_{(112)} &= {}^{c}R_{c^*(11)}\ {}^{i}t_{c^*(2)} - {}^{c}t_{c^*(1)}\ {}^{i}R_{c^*(21)} \\
\mathcal{T}_{(113)} &= {}^{c}R_{c^*(11)}\ {}^{i}t_{c^*(3)} - {}^{c}t_{c^*(1)}\ {}^{i}R_{c^*(31)} \\
\mathcal{T}_{(121)} &= {}^{c}R_{c^*(21)}\ {}^{i}t_{c^*(1)} - {}^{c}t_{c^*(2)}\ {}^{i}R_{c^*(11)} \\
\mathcal{T}_{(122)} &= {}^{c}R_{c^*(21)}\ {}^{i}t_{c^*(2)} - {}^{c}t_{c^*(2)}\ {}^{i}R_{c^*(21)} \\
\mathcal{T}_{(123)} &= {}^{c}R_{c^*(21)}\ {}^{i}t_{c^*(3)} - {}^{c}t_{c^*(2)}\ {}^{i}R_{c^*(31)} \\
\mathcal{T}_{(131)} &= {}^{c}R_{c^*(31)}\ {}^{i}t_{c^*(1)} - {}^{c}t_{c^*(3)}\ {}^{i}R_{c^*(11)} \\
\mathcal{T}_{(132)} &= {}^{c}R_{c^*(31)}\ {}^{i}t_{c^*(2)} - {}^{c}t_{c^*(3)}\ {}^{i}R_{c^*(21)} \\
\mathcal{T}_{(133)} &= {}^{c}R_{c^*(31)}\ {}^{i}t_{c^*(3)} - {}^{c}t_{c^*(3)}\ {}^{i}R_{c^*(31)} \\
\mathcal{T}_{(211)} &= {}^{c}R_{c^*(12)}\ {}^{i}t_{c^*(1)} - {}^{c}t_{c^*(1)}\ {}^{i}R_{c^*(12)} \\
\mathcal{T}_{(212)} &= {}^{c}R_{c^*(12)}\ {}^{i}t_{c^*(2)} - {}^{c}t_{c^*(1)}\ {}^{i}R_{c^*(22)} \\
\mathcal{T}_{(213)} &= {}^{c}R_{c^*(12)}\ {}^{i}t_{c^*(3)} - {}^{c}t_{c^*(1)}\ {}^{i}R_{c^*(32)} \\
\mathcal{T}_{(221)} &= {}^{c}R_{c^*(22)}\ {}^{i}t_{c^*(1)} - {}^{c}t_{c^*(2)}\ {}^{i}R_{c^*(12)} \\
\mathcal{T}_{(222)} &= {}^{c}R_{c^*(22)}\ {}^{i}t_{c^*(2)} - {}^{c}t_{c^*(2)}\ {}^{i}R_{c^*(22)} \\
\mathcal{T}_{(223)} &= {}^{c}R_{c^*(22)}\ {}^{i}t_{c^*(3)} - {}^{c}t_{c^*(2)}\ {}^{i}R_{c^*(32)} \\
\mathcal{T}_{(231)} &= {}^{c}R_{c^*(32)}\ {}^{i}t_{c^*(1)} - {}^{c}t_{c^*(3)}\ {}^{i}R_{c^*(12)} \\
\mathcal{T}_{(232)} &= {}^{c}R_{c^*(32)}\ {}^{i}t_{c^*(2)} - {}^{c}t_{c^*(3)}\ {}^{i}R_{c^*(22)} \\
\mathcal{T}_{(233)} &= {}^{c}R_{c^*(32)}\ {}^{i}t_{c^*(3)} - {}^{c}t_{c^*(3)}\ {}^{i}R_{c^*(32)} \\
\mathcal{T}_{(311)} &= {}^{c}R_{c^*(13)}\ {}^{i}t_{c^*(1)} - {}^{c}t_{c^*(1)}\ {}^{i}R_{c^*(13)} \\
\mathcal{T}_{(312)} &= {}^{c}R_{c^*(13)}\ {}^{i}t_{c^*(2)} - {}^{c}t_{c^*(1)}\ {}^{i}R_{c^*(23)} \\
\mathcal{T}_{(313)} &= {}^{c}R_{c^*(13)}\ {}^{i}t_{c^*(3)} - {}^{c}t_{c^*(1)}\ {}^{i}R_{c^*(33)} \\
\mathcal{T}_{(321)} &= {}^{c}R_{c^*(23)}\ {}^{i}t_{c^*(1)} - {}^{c}t_{c^*(2)}\ {}^{i}R_{c^*(13)} \\
\mathcal{T}_{(322)} &= {}^{c}R_{c^*(23)}\ {}^{i}t_{c^*(2)} - {}^{c}t_{c^*(2)}\ {}^{i}R_{c^*(23)} \\
\mathcal{T}_{(323)} &= {}^{c}R_{c^*(23)}\ {}^{i}t_{c^*(3)} - {}^{c}t_{c^*(2)}\ {}^{i}R_{c^*(33)} \\
\mathcal{T}_{(331)} &= {}^{c}R_{c^*(33)}\ {}^{i}t_{c^*(1)} - {}^{c}t_{c^*(3)}\ {}^{i}R_{c^*(13)} \\
\mathcal{T}_{(332)} &= {}^{c}R_{c^*(33)}\ {}^{i}t_{c^*(2)} - {}^{c}t_{c^*(3)}\ {}^{i}R_{c^*(23)} \\
\mathcal{T}_{(333)} &= {}^{c}R_{c^*(33)}\ {}^{i}t_{c^*(3)} - {}^{c}t_{c^*(3)}\ {}^{i}R_{c^*(33)}
\end{aligned} \tag{9}$$

Our objective is to move our robot from the initial camera position to the desired target camera position. Therefore, we have two special cases for the tensor coefficients values.

First when C_c at the initial position C_i

$$\begin{gathered}
{}^{c}R_{c^*} = {}^{i}R_{c^*},\ \ {}^{c}t_{c^*} = {}^{i}t_{c^*} \\
\mathcal{T}^{i}_{(jkl)} = {}^{c}R_{c^*(kj)}\ {}^{c}t_{c^*(l)} - {}^{c}t_{c^*(k)}\ {}^{c}R_{c^*(lj)} \\
\text{when } k = l, \mathcal{T}^{i}_{(jkl)} = 0 \\
\mathcal{T}^{i}_{(112)} = -\mathcal{T}^{i}_{(121)}, \mathcal{T}^{i}_{(113)} = -\mathcal{T}^{i}_{(131)}, \mathcal{T}^{i}_{(123)} = -\mathcal{T}^{i}_{(132)} \\
\mathcal{T}^{i}_{(212)} = -\mathcal{T}^{i}_{(221)}, \mathcal{T}^{i}_{(213)} = -\mathcal{T}^{i}_{(231)}, \mathcal{T}^{i}_{(223)} = -\mathcal{T}^{i}_{(232)} \\
\mathcal{T}^{i}_{(312)} = -\mathcal{T}^{i}_{(321)}, \mathcal{T}^{i}_{(313)} = -\mathcal{T}^{i}_{(331)}, \mathcal{T}^{i}_{(323)} = -\mathcal{T}^{i}_{(332)}
\end{gathered} \tag{10}$$

Second when C_c at the desired position C_{c^*}

$$\begin{gathered}
{}^{c}R_{c^*} = I,\ \ {}^{c}t_{c^*} = \mathbf{0} \\
\mathcal{T}^{*}_{(jkl)} = I_{(kj)}\ {}^{i}t_{c^*(l)} \\
\text{when } j \neq k, \mathcal{T}^{*}_{(jkl)} = 0 \\
\mathcal{T}^{*}_{(111)} = \mathcal{T}^{*}_{(221)} = \mathcal{T}^{*}_{(331)} = {}^{i}t_{c^*(1)} \\
\mathcal{T}^{*}_{(112)} = \mathcal{T}^{*}_{(222)} = \mathcal{T}^{*}_{(332)} = {}^{i}t_{c^*(2)} \\
\mathcal{T}^{*}_{(113)} = \mathcal{T}^{*}_{(223)} = \mathcal{T}^{*}_{(333)} = {}^{i}t_{c^*(3)}
\end{gathered} \tag{11}$$

3.2 Tensor Derivation

First, The derivatives of all the trifocal tensor elements with respect to time are generally as following:

$$\dot{\mathcal{T}}_{(jkl)} = {}^{c}\dot{R}_{c^*(kj)}\ {}^{i}t_{c^*(l)} + {}^{c}R_{c^*(kj)}\ {}^{i}\dot{t}_{c^*(l)} - {}^{c}\dot{t}_{c^*(k)}\ {}^{i}R_{c^*(lj)} - {}^{c}t_{c^*(k)}\ {}^{i}\dot{R}_{c^*(lj)} \tag{12}$$

Since our initial camera C_i is fixed, the elements of the derivatives of its rotation matrix and its transpose vector are equal to zero, *i.e.:* ${}^{i}\dot{t}_{c^*(l)} = {}^{i}\dot{R}_{c^*(jl)} = 0$. Our trifocal tensor elements derivative is then simplified to:

$$\dot{\mathcal{T}}_{(jkl)} = {}^{c}\dot{R}_{c^*(kj)}\ {}^{i}t_{c^*(l)} - {}^{c}\dot{t}_{c^*(k)}\ {}^{i}R_{c^*(lj)} \tag{13}$$

The spatial velocity of the camera is $u_c = (v_c, \omega_c)^T$, where v_c and ω_c are the translational and rotational velocities of the camera. From the geometry of the scene, we can deduce the following relationships:

$$[\omega_c]_\times = {}^{c^*}R_c^{T}\,{}^{c^*}\dot{R}_c = -{}^{c^*}\dot{R}_c^{T}\,{}^{c^*}R_c$$

$$
\begin{aligned}
{}^{c^*}\dot{R}_c{}^T &= -[\omega_c]_\times {}^{c^*}R_c{}^T \\
{}^{c^*}\dot{R}_c{}^T &= -[\omega_c]_\times {}^{c}R_{c^*} \\
{}^{c}\dot{R}_{c^*} &= -[\omega_c]_\times {}^{c}R_{c^*} \\
{}^{c^*}\dot{t}_c &= {}^{c^*}R_c v_c \\
{}^{c}t_{c^*} &= -{}^{c}R_{c^*} {}^{c^*}t_c \\
{}^{c}\dot{t}_{c^*} &= -{}^{c}\dot{R}_{c^*} {}^{c^*}t_c - {}^{c}R_{c^*} {}^{c^*}\dot{t}_c \\
{}^{c}\dot{t}_{c^*} &= [\omega_c]_\times {}^{c}R_{c^*} {}^{c^*}t_c - {}^{c}R_{c^*} {}^{c^*}R_c v_c \\
{}^{c}\dot{t}_{c^*} &= [\omega_c]_\times {}^{c}t_{c^*} - v_c \\
{}^{c}\dot{t}_{c^*} &= [{}^{c}t_{c^*}]_\times \omega_c - v_c \\
{}^{c}\dot{t}_{c^*(k)} &= [-I | [{}^{c}t_{c^*}]_\times] u
\end{aligned}
$$

Substituting back into the tensor derivation (13), we get:

$$
\begin{aligned}
\dot{\mathcal{T}}_{(jkl)} &= -([\omega_c]_\times {}^{c}R_{c^*})_{(kj)}\ {}^{i}t_{c^*(l)} - ([{}^{c}t_{c^*}]_\times \omega_c - v_c)_{(k)}\ {}^{i}R_{c^*(lj)} \\
\dot{\mathcal{T}}_{(jkl)} &= -(\sum_m [\omega_c]_{\times(km)} {}^{c}R_{c^*(mj)})\ {}^{i}t_{c^*(l)} - ((([{}^{c}t_{c^*}]_\times \omega_c)_{(k)} - v_{c(k)})\ {}^{i}R_{c^*(lj)}
\end{aligned}
\tag{14}
$$

where

$$
{}^{i}t_{c^*} = -{}^{i}R_{c^*} \begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix},\ {}^{c}t_{c^*} = -{}^{c}R_{c^*} \begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix},
$$

$$
{}^{i}R_{c^*} = \begin{bmatrix} R_{i(11)} & R_{i(12)} & R_{i(13)} \\ R_{i(21)} & R_{i(22)} & R_{i(23)} \\ R_{i(31)} & R_{i(32)} & R_{i(33)} \end{bmatrix},\ {}^{c}R_{c^*} = \begin{bmatrix} R_{c(11)} & R_{c(12)} & R_{c(13)} \\ R_{c(21)} & R_{c(22)} & R_{c(23)} \\ R_{c(31)} & R_{c(32)} & R_{c(33)} \end{bmatrix},
$$

$$
[\omega_c]_\times = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_y \\ -\omega_y & \omega_x & 0 \end{bmatrix},\ [{}^{c}t_{c^*}]_\times \omega_c = \begin{bmatrix} -{}^{c}t_{c^*(3)}\omega_y + {}^{c}t_{c^*(2)}\omega_z \\ {}^{c}t_{c^*(3)}\omega_x - {}^{c}t_{c^*(1)}\omega_z \\ -{}^{c}t_{c^*(2)}\omega_x + {}^{c}t_{c^*(1)}\omega_y \end{bmatrix}
$$

From (9) and (14), the derivative of the first tensor coefficient can be computed as follows:

$$
\begin{aligned}
\dot{\mathcal{T}}_{(111)} &= -(-R_{c(21)}\omega_z + R_{c(31)}\omega_y)\ {}^{i}t_{c^*(1)} - (-{}^{c}t_{c^*(3)}\omega_y + {}^{c}t_{c^*(2)}\omega_z - v_x)\ {}^{i}R_{c^*(11)} \\
\dot{\mathcal{T}}_{(111)} &= R_{i(11)}v_x - (R_{c(31)} {}^{i}t_{c^*(1)} - R_{i(11)} {}^{c}t_{c^*(3)})\omega_y + (R_{c(21)} {}^{i}t_{c^*(1)} - R_{i(11)} {}^{c}t_{c^*(2)})\omega_z \\
\dot{\mathcal{T}}_{(111)} &= R_{i(11)}v_x - \mathcal{T}_{(131)}\omega_y + \mathcal{T}_{(121)}\omega_z
\end{aligned}
\tag{15}
$$

By expanding the derivatives for the rest of the coefficients, we can deduce this compact form for the tensor coefficients derivatives:

$$
\dot{\mathcal{T}}_{(jkl)} = R_{i(lj)}v_{c(k)} - \sum_m [\omega_c]_{\times(km)} \mathcal{T}_{(jml)} \tag{16}
$$

3.3 Tensor Normalization

In this approach, no metric information, depth estimation or trifocal tensor decomposition were used. Hence, we need to define a common scale of the trifocal tensor for the control law. The trifocal tensor is normalized to get a fixed scale with $\mathcal{T}_N$. When the camera reaches the desired goal position, we can observe from (11) that the tensor coefficients are only related to the translation of the camera at the initial position which is a constant value vector. Thus, the norm of the vector $\|{}^i t_{c^*}\|$ can be used as a normalization factor for the trifocal tensor. Then the normalized trifocal tensor coefficients are:

$$
\begin{aligned}
T_{jkl} &= \frac{\mathcal{T}_{jkl}}{\mathcal{T}_N} \\
\mathcal{T}_N = \|{}^i t_{c^*}\| &= \sqrt{\left({}^i t_{c^*(1)}\right)^2 + \left({}^i t_{c^*(1)}\right)^2 + \left({}^i t_{c^*(3)}\right)^2}
\end{aligned} \tag{17}
$$

The use of the initial camera position is conveniant in visual servoing as the normalization factor $\mathcal{T}_N$ is never equal to zero except when the initial and desired camera positions are the same, which is not valid in our context. Also, this normalization factor is constant with time and doesn't affect the tensor derivatives. The trifocal tensor estimation is then computed relative to the initial camera position distance which can be estimated in the initialization step and used further for the control loop.

3.4 Interaction Matrix

To control the six degrees of freedom, at least six tensor coefficients are necessary. However, to avoid singularities, more than six tensor coefficients are considered. The interaction matrix taking into account all the tensor coefficients is given in (18). It is important to notice that pose of the initial camera location is necessary for the computation of the interaction matrix as well as the normalization factor. This initial pose can be retrieved by computing the equivalent projection matrix for the estimated trifocal tensor as show in 2.3. This step is done at the control loop initialization.

$$
L_{\mathcal{T}_{(jkl)}} = \frac{1}{\mathcal{T}_N}
\begin{bmatrix}
{}^iR_{c^*(11)} & 0 & 0 & 0 & -\mathcal{T}_{(131)} & \mathcal{T}_{(121)} \\
{}^iR_{c^*(21)} & 0 & 0 & 0 & -\mathcal{T}_{(132)} & \mathcal{T}_{(122)} \\
{}^iR_{c^*(31)} & 0 & 0 & 0 & -\mathcal{T}_{(133)} & \mathcal{T}_{(123)} \\
0 & {}^iR_{c^*(11)} & 0 & \mathcal{T}_{(131)} & 0 & -\mathcal{T}_{(111)} \\
0 & {}^iR_{c^*(21)} & 0 & \mathcal{T}_{(132)} & 0 & -\mathcal{T}_{(112)} \\
0 & {}^iR_{c^*(31)} & 0 & \mathcal{T}_{(133)} & 0 & -\mathcal{T}_{(113)} \\
0 & 0 & {}^iR_{c^*(11)} & -\mathcal{T}_{(121)} & \mathcal{T}_{(111)} & 0 \\
0 & 0 & {}^iR_{c^*(21)} & -\mathcal{T}_{(122)} & \mathcal{T}_{(112)} & 0 \\
0 & 0 & {}^iR_{c^*(31)} & -\mathcal{T}_{(123)} & \mathcal{T}_{(113)} & 0 \\
{}^iR_{c^*(12)} & 0 & 0 & 0 & -\mathcal{T}_{(231)} & \mathcal{T}_{(221)} \\
{}^iR_{c^*(22)} & 0 & 0 & 0 & -\mathcal{T}_{(232)} & \mathcal{T}_{(222)} \\
{}^iR_{c^*(32)} & 0 & 0 & 0 & -\mathcal{T}_{(233)} & \mathcal{T}_{(223)} \\
0 & {}^iR_{c^*(12)} & 0 & \mathcal{T}_{(231)} & 0 & -\mathcal{T}_{(211)} \\
0 & {}^iR_{c^*(22)} & 0 & \mathcal{T}_{(232)} & 0 & -\mathcal{T}_{(212)} \\
0 & {}^iR_{c^*(32)} & 0 & \mathcal{T}_{(233)} & 0 & -\mathcal{T}_{(213)} \\
0 & 0 & {}^iR_{c^*(12)} & -\mathcal{T}_{(221)} & \mathcal{T}_{(211)} & 0 \\
0 & 0 & {}^iR_{c^*(22)} & -\mathcal{T}_{(222)} & \mathcal{T}_{(212)} & 0 \\
0 & 0 & {}^iR_{c^*(32)} & -\mathcal{T}_{(223)} & \mathcal{T}_{(213)} & 0 \\
{}^iR_{c^*(13)} & 0 & 0 & 0 & -\mathcal{T}_{(331)} & \mathcal{T}_{(321)} \\
{}^iR_{c^*(23)} & 0 & 0 & 0 & -\mathcal{T}_{(332)} & \mathcal{T}_{(322)} \\
{}^iR_{c^*(33)} & 0 & 0 & 0 & -\mathcal{T}_{(333)} & \mathcal{T}_{(323)} \\
0 & {}^iR_{c^*(13)} & 0 & \mathcal{T}_{(331)} & 0 & -\mathcal{T}_{(311)} \\
0 & {}^iR_{c^*(23)} & 0 & \mathcal{T}_{(332)} & 0 & -\mathcal{T}_{(312)} \\
0 & {}^iR_{c^*(33)} & 0 & \mathcal{T}_{(333)} & 0 & -\mathcal{T}_{(313)} \\
0 & 0 & {}^iR_{c^*(13)} & -\mathcal{T}_{(321)} & \mathcal{T}_{(311)} & 0 \\
0 & 0 & {}^iR_{c^*(23)} & -\mathcal{T}_{(322)} & \mathcal{T}_{(312)} & 0 \\
0 & 0 & {}^iR_{c^*(33)} & -\mathcal{T}_{(323)} & \mathcal{T}_{(313)} & 0
\end{bmatrix}
\tag{18}
$$

References

[1] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, ISBN: 0521540518, 2004.

[2] T. Rowland and E. W. Weisstein, "Tensor," http://mathworld.wolfram.com/Tensor.html, last visited on 03/02/2015.

[3] C. Stover and E. W. Weisstein, "Einstein summation," http://mathworld.wolfram.com/EinsteinSummation.html, last visited on 03/02/2015.

[4] G. López-Nicolás, J. J. Guerrero, and C. Sagüés, "Visual control through the trifocal tensor for nonholonomic robots," *Robotics and Autonomous Systems*, vol. 58, no. 2, pp. 216–226, 2010.

[5] A. Shademan and M. Jagersand, "Three-view uncalibrated visual servoing," in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on.* IEEE, 2010, pp. 6234–6239.