# Visual Trajectories from Uncalibrated Stereo

Andreas Ruf

ruf@inrialpes.fr

Radu Horaud

horaud@inrialpes.fr

GRAVIR–IMAG, INRIA Rhône–Alpes, 38330 Montbonnot St. Martin, FRANCE

March 31, 1999

## Abstract

*This paper proposes a methodological framework for trajectory generation in projective space. The framework enables trajectories that respect rigidity constraints of structure and motion, that are furthermore uniform and optimal in terms of Euclidean configuration, eventhough no camera calibration is assumed.*

*In a projective reconstruction of the scene, obtained e.g. from an uncalibrated stereo-rig, the homography between corresponding feature points on a rigid object in different positions is employed to represent its spatial configuration. From its algebraic structure - it is similar to a rigid motion - decoupling of rotation and translation, and uniform interpolation of trajectories are derived.*

*Speaking in robotic terms, the merely point-to-point definitions of goals, as present in current approaches to visual servoing, are extended to a guarded motion along via points on complete-path trajectories which result from interpolation of a demanded type of motion. Translation-only, rotation-first, rotation-last task-functions are further explicited.*

# Introduction and motivation

Computer vision in robotics started as off-line system for part- or landmark-recognition, trajectory planning or mobile robot navigation, and computer-aided quality control. From the continuing increase in the capabilities of these systems and especially from the decrease of run-time towards real-time, the integration of computer vision into the on-line loop has become feasible. In consequence, the domain of vision systems has extended to object detection, object tracking and pose estimation for the purposes of obstacle avoidance, target tracking, execution monitoring, and finally for position-based control. The respective control laws run an open loop which consists of successive but well separated stages, one for determining the position of the manipulator and one for correcting this position by actuator motions towards the control objective. These approaches are classified as “look-and-move” [6].

## 1.1 Closed-loop image-based approach

Further reduction of cycle time nowadays allows a full integration of dynamic visual sensor input into closed-loop feedback control [3]. In contrast to open-loop control, a closed-loop is less sensitive to inaccuracies in calibration, is robust to disturbances during execution, requires however careful design of its dynamics in order to ensure convergence and stability [12]. Indeed, such dynamic visual servoing systems have recently been developed. They are called ”image-based”, since the target is given merely in terms of images, e.g. by a set of image points, since the error-function is a distance between the current and the desired configuration of these image points, and since the control-law reduces this error by servoing a robot in closed-loop mode without explicitely determining pose or spatial structure [5]. For these control-laws, local convergence and stability in presence of calibration errors has been demonstrated [2]. The principal deficiencies are that these approaches enable only point-to-point goals, that trajectories are not further confined, particularly that they are not optimal, that vanishing features due to occlusion or loss of focus often result in divergence, and that even in case of complete visibility convergence is guaranteed only locally. All these are consequences of the fact that rigidity of structure and motion is no longer ensured when calculations are solely image-based. More figuratively, the straight-forward descend on the gradient of image error in P-controllers does not “look-ahead”. So, either local minima might trap the controller before reaching the final goal or sudden changes in error due to occlusion or loss of focus cause divergence. Furthermore, trajectories induced by image-based gradient descend are neither physically valid nor optimal, i.e. they tend to deform structure by violating rigidity of motion and additionally fail to constrain motion to the shortest path towards the goal and direction of approach.

## 1.2 Generic task-functions

In this paper, we propose a method for complete-path goal definition and subsequent generation of optimal trajectories for image-based visual servoing. Although just weakly calibrated stereo-rigs are assumed, trajectories respect rigidity and guarantee shortest path-to-goal. At first, trajectories

are calculated in projective space from a spatial reconstruction of the scene. They are later reprojected to the images to serve as set-points for image-based control.

The investigated classes of complete-path trajectories include translation-only, parallel-translation, rotation-first, rotation-last trajectories. A Cartesian-move mode in projective space should also be feasible. These occur quite often in pratice. For instance, the list of orders given by an operator in video-based telemanipulation constitutes generic robotic tasks that correspond to certain trajectory classes:

- grasp object, e.g. fix, load part
- lift object, e.g. car disassembly
- put-down object on table, e.g. surface mount
- insert peg into hole, e.g. plug in socket
- turn object, e.g. turn valve
- drag object, e.g. move heavy load
- parallel alignment of tool to target, e.g. driver to screw
- perpendicular alignment of tool and planar target, e.g. drill normal to surface
- angular alignment of tool and target, e.g. spray-can normal to surface

### 1.3 Outline

The paper is divided into six sections. After the introduction in section 1, some algebraic and geometric fundamentals are laid down in section 2. In section 3 the representation of pure translations in a projective frame is investigated. An algebraic algorithm for translation-based affine auto-calibration follows in section 4, including some consequences on the generation of point-to-point translations and R-T decomposition, both in a projective frame with known affine properties. The relevance of the preceeding results for visual servoing is demonstrated in section 5, which develops the continuous, uniform, complete-path task-function for a straight-line translation, observed by weakly calibrated cameras. Section 6 concludes with a discussion of the presented framework.

## 2 Fundamentals

### 2.1 Extrinsic geometry

The geometry of the scene is given in a world-frame of reference $\mathcal{W}$ and points $\boldsymbol{X}$ in $\mathcal{W}$ are in homogeneous coordinates ${}_W\boldsymbol{X} = [X, Y, Z, 1]^T$. A Euclidean camera-frame $\mathcal{C}$ is canonically attached to each camera, such that the origin of $\mathcal{C}$ coincides with the optical center, the $z$-axis coincides with the optical axis, the $x$-$y$ plane becomes the focal plane, parallel to the image plane, and the $y$-axis is parallel to image scan-lines (Fig. 1). The displacement between world- and camera-frame is written as homogeneous transform $\boldsymbol{T_{RT}}$, which maps $\mathcal{W}$-coordinates onto $\mathcal{C}$-coordinates: ${}_C\boldsymbol{X} = \boldsymbol{T_{RT}}\ {}_W\boldsymbol{X}$.

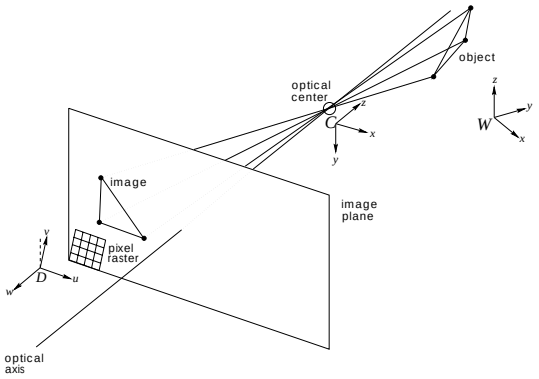


Figure 1: Pin-hole camera model.

### 2.2 Intrinsic geometry

A canonical affine frame $\mathcal{D}$ is attached to the video-camera (CCD), such that the $u$-$v$-origin is the pixel origin, the $u$-axis is parallel to pixel columns, the $v$-axis is again parallel to image scan-lines, and the $w$-axis is parallel to the optical axis with unit length $f$ in negative $z$-direction. Units on the $u$- and $v$-axis are in pixel of width $1/k_u$ and of height $1/k_v$. The transition from camera-frame $\mathcal{C}$ to video-frame $\mathcal{D}$ is by the affine transform $\boldsymbol{A}$ (Fig. 1).

$$\begin{bmatrix} \gamma U \\ \gamma V \\ \gamma W \\ \gamma \end{bmatrix} = \begin{bmatrix} k_u & k_{uv} & -u_0/f & 0 \\ 0 & k_v & -v_0/f & 0 \\ 0 & 0 & -1/f & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{R} & \boldsymbol{t} \\ 0\ 0\ 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix},$$

$$\gamma \boldsymbol{N} = \boldsymbol{A}\ \boldsymbol{T_{RT}}\ \boldsymbol{X}, \tag{1}$$

The deviation of the $u$-axis from orthogonality to $v$ introduces a affine skew in the focal plane by $k_{uv}$.

### 2.3 Image projection

In $\mathcal{D}$, normalization with $\gamma$ yields affine space coordinates $[U, V, W]^T$ of $\boldsymbol{N}$. Normalizing with $\gamma W$ results in image coordinates $[u, v]^T$ in pixel.

$$\begin{bmatrix} \gamma U \\ \gamma V \\ \gamma W \\ \gamma \end{bmatrix} = \gamma \begin{bmatrix} \begin{bmatrix} U \\ V \\ W \end{bmatrix} \\ 1 \end{bmatrix} = \gamma W \begin{bmatrix} \frac{U}{W} \\ \frac{V}{W} \\ 1 \\ \frac{1}{W} \end{bmatrix} = \gamma W \begin{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} \\ 1 \\ \frac{1}{W} \end{bmatrix} \tag{2}$$

We will revisit this purely scalar relation between image coordinates and affine space coordinates in $\mathcal{D}$ in section 5. The usual formulation of perspective projection evokes now as a special case of the representation in the affine frame $\mathcal{D}$, mainly because the $u$-$v$-plane is parallel to the focal plane (Fig. 1) .

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha_u & \alpha_{uv} & u_0 & 0 \\ 0 & \alpha_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \boldsymbol{T_{RT}}\ \boldsymbol{X},$$

$$\boldsymbol{x} = \boldsymbol{P_E}\ \boldsymbol{X}, \tag{3}$$

with $\alpha_u = -fk_u$, $\alpha_v = -fk_v$: $(u_0, v_0)$ is the pixel where the optical axis hits the image plane, and $\alpha_{uv}$ describes the affine skew between columns and rows of the CCD: the five intrinsic parameters of the camera.

### 2.4 Projective reconstruction

It is well know, that from at least two uncalibrated images and eight common points, a reconstruction of the scence in a freely chosen frame $\mathcal{R}$ of projective three-space $P(3)$ can be calculated [4], [11]. They are however a projectivity $\boldsymbol{S_E}$ away from a Euclidean reconstruction in $\mathcal{W}$

$$\begin{array}{rcccl} \gamma \boldsymbol{N} & = & \boldsymbol{A}\, \boldsymbol{S_E} & \boldsymbol{S_E}^{-1}\, \boldsymbol{X}, \\ \gamma \boldsymbol{N} & = & \boldsymbol{H_A} & \boldsymbol{M}, \end{array} \tag{4}$$

where $\boldsymbol{M}$ is the representation of $\boldsymbol{X}$ in $\mathcal{R}$ and $\boldsymbol{S_E}$ is a "Euclideanizing" homography $\boldsymbol{X} = \boldsymbol{S_E M}$.

### 2.5 Projective displacements

Consider now a set of $k$ points $\boldsymbol{X_i}$ on a rigid body and let this body undergo a rigid displacement $\boldsymbol{T_{RT}}$, such that $\boldsymbol{X_i'} = \boldsymbol{T_{RT} X_i}$. Changing to projective coordinates

$$\begin{array}{rcccl} \boldsymbol{S_E}^{-1} \boldsymbol{X_i}' & = & \boldsymbol{S_E}^{-1} \boldsymbol{T_{RT}} \boldsymbol{S_E} & \boldsymbol{S_E}^{-1} \boldsymbol{X_i} \\ \boldsymbol{M_i'} & = & \gamma \quad \boldsymbol{H_{RT}} & \boldsymbol{M_i} \end{array} \tag{5}$$

shows that the Euclidean representation $\boldsymbol{T_{RT}}$ and the projective representation $\boldsymbol{H_{RT}}$ of one and the same displacement are related by a scaled algebraic similarity with $\boldsymbol{S_E}$ as similarity transform (see also [13], [1])

$$\boldsymbol{H_{RT}} = \gamma \boldsymbol{S_E}^{-1} \boldsymbol{T_{RT}} \boldsymbol{S_E}. \tag{6}$$

Call "projective displacements" the subgroup of projective transforms, which are similar to a rigid displacement (Fig. 2).

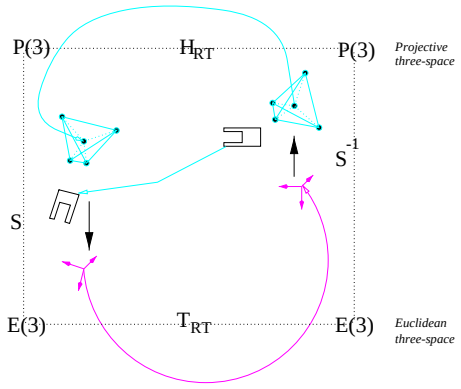


Figure 2: Euclideanizing homography $\boldsymbol{S_E}$ and the resulting similarity between a rigid $\boldsymbol{T_{RT}}$ and a projective displacement $\boldsymbol{H_{RT}}$.

### 2.6 Plane at infinity

A point is said to be "at infinity" iff its affine coordinates fulfill $\boldsymbol{N} = [U, V, W, 0]^T$. The set of all points at infinity form a "plane at infinity" $\Pi_\infty$, which is defined by the plane equation $\boldsymbol{\Pi_\infty}^T \boldsymbol{N} = 0$, so $\boldsymbol{\Pi_\infty} = [0, 0, 0, 1]^T$ in affine coordinates. Projective coordinates of $\Pi_\infty$ follow from applying the infinity constraint to rectified coordinates $\boldsymbol{N} = \boldsymbol{S_E M}$

$$[0, 0, 0, 1]^T \boldsymbol{S_E M} = 0 \text{ iff } \boldsymbol{S_{E(4,\cdot)}}^T \boldsymbol{M} = 0, \tag{7}$$

which reveals the forth row $\boldsymbol{S_{E(4,\cdot)}}^T$ of $\boldsymbol{S_E}$ to hold the coordinates of $\Pi_\infty$ in $\mathcal{R}$. $\boldsymbol{\Pi_\infty}$ rests fix under projective displacements, i.e. it is an eigenvector of any $\boldsymbol{H}_{RT}^{-T}$. Hence, its coordinates depend mainly on the choice of the projective frame in which the scene was reconstructed.

### 2.7 Canonic representation

The so far arbitrarily fixed Euclidean world-frame can be constrained in a canonical way, as it is done by the "Jordan canonical form"[1] (*JCF* ) of similar matrices. In case of a projective displacement, the similarity class writes as

$$\begin{array}{rcl} \boldsymbol{H_{RT}} & = & \gamma \boldsymbol{S_{RT}}^{-1} \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \boldsymbol{S_{RT}}, \\ \boldsymbol{H_{RT}} & = & \gamma \boldsymbol{S_{RT}}^{-1} \boldsymbol{J_{RT}} \boldsymbol{S_{RT}} \end{array} \tag{8}$$

where the canonical frame is such that rotation is by an angle $\theta$ around the $z$-axis and translation is of unit length in $z$-direction. The $x$- and $y$-axis are chosen mutually orthogonal in a plane perpendicular to the $z$-axis. This corresponds to representing $\boldsymbol{H_{RT}}$ in the frame of normalized screw-motion, i.e. the origin is shifted to lie on the screw-axis, which then coincides with the $z$-axis, and scales are such that the screw-pitch is of unit-length (Fig. 3). The algebraic characterization of $\boldsymbol{H_{RT}}$ is obvious from its *JCF*. In the general case, the only real eigenvalue is $\gamma$, with algebraic multiplicity two, but geometric multiplicity one. The complex eigenvalue pair is $(\gamma e^{i\theta}, \gamma e^{-i\theta})$, where $\theta$ equals the angle of rotation. A choice of eigenvectors are the columns of $\boldsymbol{S_{RT}}^{-1}$. The first two span an orthonormal basis of the $x$-$y$-plane, the third is parallel to the $z$-axis. Together they form an orthogonal triad spanning Euclidean vector space.

## 3 Pure translations

By the *JCF* some special similarity classes of $\boldsymbol{H_{RT}}$ are identified. A *projective translation* $\boldsymbol{H_T}$ is a projective transform, which has a *JCF* $\boldsymbol{J_T}$ defined like

$$\begin{array}{rcl} \boldsymbol{H_T} & = & \gamma\, \boldsymbol{S_T}^{-1} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \boldsymbol{S_T} \\ & = & \gamma\, \boldsymbol{S_T}^{-1} \boldsymbol{J_T} \boldsymbol{S_T}, \end{array} \tag{9}$$

[1] The Jordan form is canonical up to permutations of the Jordan blocks. Without loss of generality, we concentrate on the permutation as given above.

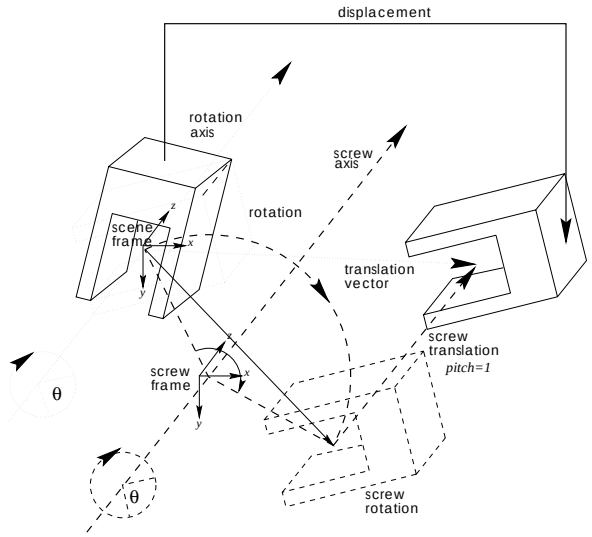


Figure 3: Screw-geometry of a rigid displacement of a gripper (solid lines). The displacement in the original frame is drawn in dotted lines, the respective screw displacement in dashed lines.

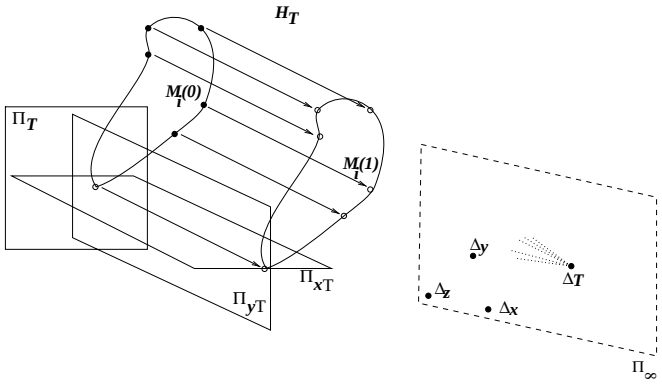


Figure 4: Projective translation $\boldsymbol{H_T}$, mapping points $\boldsymbol{M_i}(0)$ onto $\boldsymbol{M_i}(1)$. The eigenspace of the point-translation $\boldsymbol{H_T}$ spans the plane at infinity $\boldsymbol{\Pi_\infty}$, e.g. with the vanishing points of coordinate axis $\boldsymbol{\Delta_x}$, $\boldsymbol{\Delta_y}$, $\boldsymbol{\Delta_z}$ as basis. The eigenspace of the plane-translation $\boldsymbol{H_T}^{-T}$ is spanned by the plane at infinity and two planes parallel to the translation, e.g. $\boldsymbol{\Pi_\infty}$, $\boldsymbol{\Pi_{xT}}$, $\boldsymbol{\Pi_{yT}}$ and the vanishing point of the translation $\boldsymbol{\Delta_T}$ is the intersection of these three planes.

where $\gamma$ is an arbitrary scale factor. $\boldsymbol{S_T}$ rectifies projective coordinates, such that translation is of unit length in $z$-direction.

Algebraically $\boldsymbol{J_T}$ is characterized by $det(\boldsymbol{J_T}) = 1$, $trace(\boldsymbol{J_T}) = 4$, the quadruple eigenvalue $\lambda = 1$, and the $dim = 3$ eigenspace $E_1 = [x, y, z, 0]^T$, which spans pointwise the plane at infinity ${\boldsymbol{\Pi_\infty}}^T = [0, 0, 0, 1]$, which is at the same time the eigenspace's orthogonal complement ${\boldsymbol{\Pi_\infty}}^T E_1 = 0$.

From duality, the plane transformation corresponding to $\boldsymbol{H_T}$ follows as

$$ {\boldsymbol{H_T}}^{-T} = 1/\gamma \, {\boldsymbol{S_T}}^{-T} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix} {\boldsymbol{S_T}}^{T}, \qquad (10) $$

Now, the dual eigenspace $E_1^{-T} = [x, y, 0, t]^T$ is the subspace of planes parallel to the translation, including the plane at infinity, and the orthogonal complement $\boldsymbol{N_\Delta} = [0, 0, 1, 0]^T$ is the direction of translation.

The translation direction is in the point eigenspace $E_1$ as well as the plane at infinity is in the plane eigenspace $E_1^{-T}$. Although they cannot be distinguished from any other vector of the eigenspace, their property of being orthogonal complement of the respective dual eigenspace allows their calculation from a given $\boldsymbol{H_T}$. We will propose an efficient algorithm for doing so in section 4.

The similarity of $\boldsymbol{H_T}$ to $\boldsymbol{J_T}$ by $\boldsymbol{S_T}$ implies directly $det(\boldsymbol{H_T}) = \gamma$, $trace(\boldsymbol{H_T}) = 4\gamma$, a quadruple eigenvalue $\lambda = \gamma$, and a transformed eigenspace $E_\gamma = {\boldsymbol{S_T}}^{-1}[x, y, z, 0]^T$. Analogous arguments apply to the dual transform.

In the following, we introduce a reduced parameterization for a projective translation. Based on this, the geometric interpretations of the parameter vectors follow easily and a factorization-based algorithm for affine autocalibration of a stereo-rig is derived. Then, a way to calculate the projective translation between a given point pair and the decomposition of a general projective displacement into rotation and translation is shown.

After that, the task-function of translational motion is explicited. The often found point-to-point definition of a positioning task is extended to a rigid translation of the complete object along a continuous-path, which yields a globally admissible task-function in the image domain. The uniform, shape-preserving interpolation of image trajectories and the incorporation of a-priori velocity profiles are finally developed.

## 3.1 Reduced parameterization

In the new form, the number of parameters of $\boldsymbol{H_T}$ reduces to eight, four of them are fix for all translations, the remaining four characterize the present translation, plus an arbitrary scale factor.

After writing the columns of ${\boldsymbol{S_T}}^{-1}$ as $\boldsymbol{c_i}$, the rows of $\boldsymbol{S_T}$ as ${\boldsymbol{r_i}}^T$, and separating the identity matrix $\boldsymbol{I}$, equation (10) becomes:

$$ \begin{aligned} \boldsymbol{H_T} &= \gamma \left( {\boldsymbol{S_T}}^{-1} \left( I + \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \right) \boldsymbol{S_T} \right) \qquad (11) \\ &= \gamma \left( \boldsymbol{I} + \begin{bmatrix} \boldsymbol{c_1} & \boldsymbol{c_2} & \boldsymbol{c_3} & \boldsymbol{c_4} \end{bmatrix} \boldsymbol{E_{34}} \begin{bmatrix} {\boldsymbol{r_1}}^T \\ {\boldsymbol{r_2}}^T \\ {\boldsymbol{r_3}}^T \\ {\boldsymbol{r_4}}^T \end{bmatrix} \right) \\ &= \gamma \left( \boldsymbol{I} + \boldsymbol{c_3} \, {\boldsymbol{r_4}}^T \right) \qquad (12) \\ &= \gamma \left( \boldsymbol{I} + \boldsymbol{H_0} \right). \qquad (13) \end{aligned} $$

The vectors $\boldsymbol{c_3}$ and $\boldsymbol{r_4}$ however are not fully independant: $trace(\boldsymbol{H_T}) = 4\gamma$ follows from similarity, and consequently $trace(\boldsymbol{H_0}) = 0$. This can likewise be seen from

$$\boldsymbol{r_4}^T \boldsymbol{c_3} = 0, \tag{14}$$

since $\boldsymbol{S_T}\boldsymbol{S_T}^{-1} = 0$. Hence, $\boldsymbol{H_0}$ is a "zero"-matrix in the sense that it has rank 1 and trace 0.

### 3.2 Plane at infinity

As shown in section 2.6, $\boldsymbol{r_4}$ is the plane at infinity $\Pi_\infty$ in the projective frame $\mathcal{R}$. Multiplying (12) from the left $\boldsymbol{r_4}^T \boldsymbol{H_T} = \gamma\, \boldsymbol{r_4}^T$ shows that $\boldsymbol{r_4}^T$ is a left eigenvector of $\boldsymbol{H_T}$, i.e. $\boldsymbol{r_4}$ is eigenvector of $\boldsymbol{H_T}^{-T}$ to eigenvalue $1/\gamma$. In geometric terms, $\boldsymbol{r_4}$ is a fixed plane of the plane-transform $\boldsymbol{H_T}^{-T}$. What is more, $\boldsymbol{r_4}$ is a plane of fixed points, as is easily seen by multiplying (12) from the right by a point $\boldsymbol{M}$ in this plane, i.e. by a point which satisfies $\boldsymbol{r_4}^T \boldsymbol{M} = 0$.

### 3.3 Vanishing point

An endlessly continued translation of an affine point $\boldsymbol{N}(t)$ along direction vector $\Delta = [u, v, w]^T$ results at the limit in the "vanishing point" $\boldsymbol{N}_\Delta$, which corresponds to the direction of translation

$$\boldsymbol{N}_\Delta := \lim_{t\to\infty} \boldsymbol{N}(t) = \lim_{t\to\infty} \begin{bmatrix} U + t\,u \\ V + t\,v \\ W + t\,w \\ 1 \end{bmatrix} = \begin{bmatrix} u \\ v \\ w \\ 0 \end{bmatrix}. \tag{15}$$

It obviously lies in the plane at infinity $\boldsymbol{Pi_\infty}^T \boldsymbol{N}_\Delta = 0$. Its image $\boldsymbol{P}\ \boldsymbol{N_\Delta}$ is called "focus of expansion" of the given translation.

In the affine frame of (10), translation is along the $z$-axis and the vanishing point becomes $\boldsymbol{N}_\Delta = [0, 0, 1, 0]^T$. Going back to the projective frame $\mathcal{R}$ by $\boldsymbol{S_T}^{-1}$ shows that the column $\boldsymbol{c_3}$ appearing in (12) holds the projective coordinates $\boldsymbol{M_\Delta}$ of the vanishing point, which characterizes projective translations along this direction, independant of their amplitude. More formally, $\boldsymbol{c_3}$ is an eigenvector of $\boldsymbol{H_T}$, which is evident by multiplying (12) from the right.

To sum up, a projective translation $\boldsymbol{H_T}$ depends merely on a scalar factor $\gamma$ and two vectors $\boldsymbol{r_4}$ and $\boldsymbol{c_3}$, which appear as a row of the rectifying homography and a column of its inverse, respectively. They obey a single constraint $\boldsymbol{r_4}^T\ \boldsymbol{c_3} = 0$. In geometric terms, $\boldsymbol{c_3}$ is the direction of translation, $\boldsymbol{r_4}$ is the plane at infinity, and the constraint places $\boldsymbol{c_3}$ at infinity, i.e. it is the limit of all points translating in this direction. The vanishing point $\boldsymbol{c_3}$ is eigenvector of the point-translation $\boldsymbol{H_T}$, the plane at infinity $\boldsymbol{r_4}$ is eigenvector of the plane-translation $\boldsymbol{H_T}^{-T}$. The eigenspace of $\boldsymbol{H_T}$ is the plane at infinity and $\boldsymbol{H_T}$ leaves all vanishing points unchanged. Dually, the eigenspace of $\boldsymbol{H_T}^{-T}$ is plane-wise fix and consists of all planes parallel to the translation, including the plane at infinity.

## 4 Affine calibration

### 4.1 The algorithm

The relationship in equation (11) gives raise to a subspace method for affine auto-calibration of perspective cameras from one or more translation observed with a weakly calibrated stereo system. In theory, affine calibration corresponds to determining the plane at infinity, which appears in $\boldsymbol{H_0}$ calculated like

$$\boldsymbol{H_0} = \boldsymbol{H_T}/\gamma - I, \tag{16}$$

where $\gamma = trace(\boldsymbol{H_T})/4$. In each row the plane at infinity $\boldsymbol{r_4}$ appears scaled by the components of $\boldsymbol{c_3}$, and likewise the direction $\boldsymbol{c_3}$ appears in each column scaled by the components of $\boldsymbol{r_4}$.

In practice, an estimation $\boldsymbol{H_0}'$ will be disturbed from fuzzy data and round-off errors. Straight-forward accumulation by componentwise sums (17) and componentwise products (18) is potentially unstable.

$$\boldsymbol{r_4} = \frac{1}{4}\sum_{i=1}^{4} \begin{bmatrix} \boldsymbol{H_0}(i,1) \\ \boldsymbol{H_0}(i,2) \\ \boldsymbol{H_0}(i,3) \\ \boldsymbol{H_0}(i,4) \end{bmatrix} \tag{17}$$

$$\boldsymbol{r_4} = \prod_{i=1}^{4} \begin{bmatrix} \boldsymbol{H_0}(i,1) \\ \boldsymbol{H_0}(i,2) \\ \boldsymbol{H_0}(i,3) \\ \boldsymbol{H_0}(i,4) \end{bmatrix}^{1/4} \tag{18}$$

As $\boldsymbol{c_3}$ might have components close to zero, numerical effacements in (17) or instabilities due to sign changes in (18) are likely to occur. This effect has likewise been observed when calculating $\det(\boldsymbol{H_T})$ instead of $trace(\boldsymbol{H_T})$ to determine $\gamma$. Even in presence of minor disturbances, the determinant became unstable.

Unlike the ill-conditioned methods above, we propose to exploit $rank(\boldsymbol{H_0}) = 1$ in the singular value decomposition (*SVD*) of $\boldsymbol{H_0}$

$$\boldsymbol{H_0}' = \boldsymbol{U}^T diag(\sigma_1, \sigma_2, \sigma_3, \sigma_4)\boldsymbol{V}, \tag{19}$$

with $\boldsymbol{U}^T = (\boldsymbol{u_1}, \boldsymbol{u_2}, \boldsymbol{u_3}, \boldsymbol{u_4})$, $\boldsymbol{V}^T = (\boldsymbol{v_1}, \boldsymbol{v_2}, \boldsymbol{v_3}, \boldsymbol{v_4})$, and the singular values in descending order. The rank constraint permits to neglect $\sigma_2$, $\sigma_3$, and $\sigma_4$, and $\boldsymbol{H_0}$ becomes

$$\boldsymbol{H_0} = \sigma_1 \boldsymbol{u_1}\boldsymbol{v_1}^T, \tag{20}$$

while at the same time minimal error $||\boldsymbol{H_0'} - \boldsymbol{H_0}||$ in the sense of the matrix norms $||\cdot||_2$ and $||\cdot||_F$ is guaranteed.

Comparison with (11) reveals

$$\boldsymbol{c_3} = \boldsymbol{u_1}, \tag{21}$$

$$\boldsymbol{r_4} = \boldsymbol{v_1}. \tag{22}$$

Singular value decomposition is equally applicable for "stacked" $\boldsymbol{H_0}^{(4n)\times n}$, which combines different $\boldsymbol{H_T}$ estimates from various translational motions expressed in the same projective base. In practice, the calibration procedure is as follows:

1. Take $n$ image pairs during arbitrary translations of the camera system and the scene.
2. Calculate a projective reconstruction of the translating points in one and the same projective base for consecutive image pairs.
3. Estimate $n$ projective displacements $\boldsymbol{H'_T}$ and calculate $\boldsymbol{H'_0}^{(4n\times n)}$ (16).
4. Calculate the *SVD* of $\boldsymbol{H'_0}^{(4n\times n)}$ (19)
5. Approximate $\boldsymbol{H_0}^{(4n\times n)}$ by (20)
6. Extract the common plane at infinity (22) and the direction of each translation (21).

## 4.2 Point-to-point translation

We are now interested in computing a projective translation $\boldsymbol{H_T}$ that maps a point $\boldsymbol{M_0}$ to a point $\boldsymbol{M_1}$:

$$\boldsymbol{M_1} = \gamma \boldsymbol{H_T} \boldsymbol{M_0} \tag{23}$$

Translation imposes parallelism on all point traces. This is a concept of affine geometry, hence, the plane at infinity will come in. Solving for $\boldsymbol{c_3}$ with (12) yields

$$\boldsymbol{c_3} = \frac{\gamma \boldsymbol{M_1} - \boldsymbol{M_0}}{\boldsymbol{r_4}^T \boldsymbol{M_0}} \tag{24}$$

The choice of $\gamma$ has to guarantee (14), so

$$\gamma = \frac{\boldsymbol{r_4}^T \boldsymbol{M_0}}{\boldsymbol{r_4}^T \boldsymbol{M_1}} \tag{25}$$

and $\boldsymbol{H_T}$ as a function of $\boldsymbol{M_0}$, $\boldsymbol{M_1}$, and $\boldsymbol{r_4}$ becomes

$$\boldsymbol{H_T} = \frac{\boldsymbol{r_4}^T \boldsymbol{M_0}}{\boldsymbol{r_4}^T \boldsymbol{M_1}} \left( \boldsymbol{I} + \frac{\boldsymbol{M_1} \boldsymbol{r_4}^T}{\boldsymbol{r_4}^T \boldsymbol{M_1}} - \frac{\boldsymbol{M_0} \boldsymbol{r_4}^T}{\boldsymbol{r_4}^T \boldsymbol{M_0}} \right) \tag{26}$$

Now, considering the calculated projective translation as a robotic task, the whole object is moved rigidly in the scene and contact, collision, or occlusion could be checked (Fig. 5).

## 4.3 R-T Decomposition

We will show how to decompose a projective displacement with respect to a central point, which moreover constrains the rotation axis to pass through it. One could think of generating rotation-first, rotation-last trajectories and a Cartesian-move mode.

An arbitrary projective displacement $\boldsymbol{H_{RT}}$ is expressed as a sequence of one projective rotation $\boldsymbol{H_R}$ and one projective translation $\boldsymbol{H_T}$ (Fig. 6):

$$\boldsymbol{H_{RT}} = \boldsymbol{H_R} \boldsymbol{H_T} \tag{27}$$

For a given center-point $\boldsymbol{M_0}$ – this might be an ideal point or an observed point – the destination $\boldsymbol{M_1}$ is calculated by

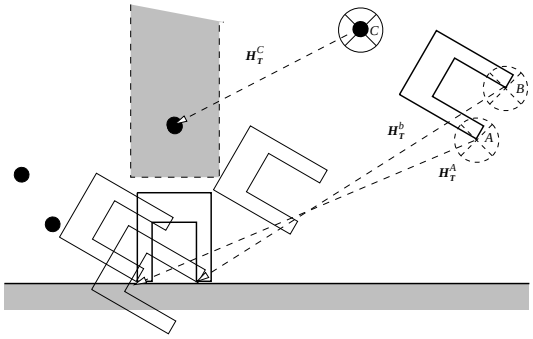


Figure 5: For an put-on-table task, a point-to-point translation is depicted for a center point close and far from the contact point, and for the wrist-center-point. Appropriate trajectory generation allows to avoid collision with the table or to decouple the control law.

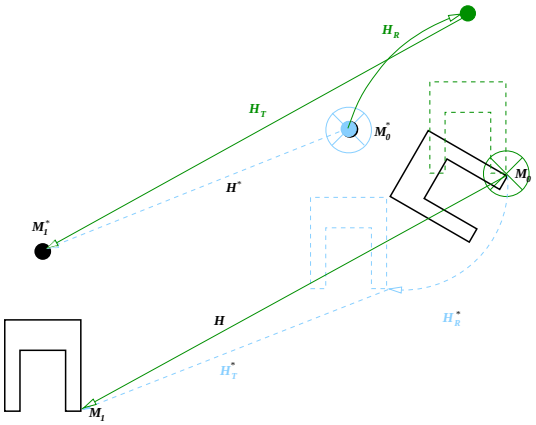

Figure 6: RT-Decomposition depending on the chosen center of rotation.

$\boldsymbol{M_1} = \boldsymbol{H_{RT}} \boldsymbol{M_0}$. The corresponding projective translation $\boldsymbol{H_T}$ is determined following section 4.2. The residual displacement $\boldsymbol{H'} = \boldsymbol{H_{RT}} {\boldsymbol{H_T}}^{-1}$ is clearly the projective representation of a pure rotation $\boldsymbol{H_R}$. To obtain a rotation-first task, decomposition is $\boldsymbol{H'} = {\boldsymbol{H_T}}^{-1} \boldsymbol{H_{RT}}$ and $\boldsymbol{H'} = \boldsymbol{H_{RT}} {\boldsymbol{H_T}}^{-1}$ for a rotation-last task. The condition $\boldsymbol{M_1} = \boldsymbol{H'} \boldsymbol{M_1}$ obviously holds, so the axis of rotation passes through $\boldsymbol{M_1}$.

In summary, affine calibration by determining the plane at infinity permits calculating a projective transform which rigidly moves a given point to a destination by a translational displacement. It furthermore enables to decompose a general displacement into a translation $\boldsymbol{H_T}$ and rotation $\boldsymbol{H_R}$, whose axis is constrained to pass through the distinguished point. This is of particular importance in robotics, as constraining the rotation to be around a point of the robots' kinematics, e.g. the robot-wrist's center point, allows to decouple the control-law.

# 5 The task of translation

We are now considering a robotic task that consists of a uniform translational displacement of an object. Given the reconstruction of a point of reference $\boldsymbol{Q}$ in its initial position $\boldsymbol{Q}(0)$, and in its target position $\boldsymbol{Q}(1)$, the corresponding projective translation $\boldsymbol{H_T}$ is determined by (26) and fully defines the task. The direction becomes $\boldsymbol{M_\Delta} = \boldsymbol{c_3}$, according to (24). Distance and orientation are implied by the target point $\boldsymbol{Q}(1)$. If translation is to be defined by its direction only, taking an arbitrary via-point as $\boldsymbol{Q}(1)$ suffices and extrapolation yields the motion beyond this point. Noteworthy, $\Delta$ is an unoriented direction only, while orientation is defined by the order of $\boldsymbol{Q}(0)$ and $\boldsymbol{Q}(1)$. In addition, the reference point is not required to lie on the object itself, i.e. any pair of points defines a translational task.

## 5.1 Singleton set-point definition

Consider $k$ points $\boldsymbol{M_i}$ on the object and their initial positions $\boldsymbol{M_i}(0)$ in the reconstruction. Compose the configuration matrix $\boldsymbol{S}(\cdot)$

$$\boldsymbol{S}(0) = \begin{pmatrix} \boldsymbol{M_1}(0)\ \boldsymbol{M_2}(0) \cdots \boldsymbol{M_k}(0) \end{pmatrix} \tag{28}$$

which represents pose and shape of the object. Apply $\boldsymbol{H_T}$ to obtain the set-point configuration $\boldsymbol{S}^*(1) = \boldsymbol{H_T}\boldsymbol{S}(0)$. Since $\boldsymbol{H_T}$ preserves rigidity, $\boldsymbol{S}^*(1)$ is, up to occlusion, an admissible configuration, i.e. corresponds to the image of the rigid object in the respective pose. The image configuration $\boldsymbol{s}$ and the image set-points $\boldsymbol{s}^*$ follow by simple back-projection

$$\boldsymbol{s}(0) = \boldsymbol{P}\ \boldsymbol{S}(0), \boldsymbol{s}^*(1) = \boldsymbol{P}\ \boldsymbol{S}^*(1), \tag{29}$$

where $\boldsymbol{P}$ is the projection matrix (3), that is available for each observing camera from the reconstruction step.

So far, this corresponds to the usual task definition by "point-point-incidence", present in known approaches to visual servoing [3]. Neither trajectory, nor direction of approaching are defined. Unlike existing approaches, the definition of the task by the projective translation $\boldsymbol{H_T}$ allows for the generation of multi-point set-points for an arbitrary rigid object. At the same time, the translation task $\boldsymbol{H_T}$ can be calculated from any observed translation vector.

## 5.2 Complete-path task-function

The definition of the task given by the projective translation $\boldsymbol{H_T}$ now is extended to a continuous-path, uniform, straight-line translation. This completely constrains the trajectory and additionally the direction of approach. Having (10) in mind, the definition [10] of our continuous task-function $\boldsymbol{H_T}(t)$ becomes

$$\boldsymbol{H_T}(t) = \gamma \boldsymbol{S_T}^{-1} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & t \\ 0 & 0 & 0 & 1 \end{bmatrix} \boldsymbol{S_T}, \quad t \in [0,1]. \tag{30}$$

Please note that uniformity is with respect to spacial Euclidean distance, i.e. $\boldsymbol{H_T}(t)$ corresponds to a $t$-fractile of the task's total distance. The formulation in (30) equally applies for moving backwards $t < 0$, towards the set-point $0 < t < 1$, and beyond the set-point $t > 1$. In contrast to simple interpolation of image set-points, rigidity is guaranteed at any $t$.

## 5.3 Uniform set-point interpolation

Thanks to the parameterization in (13), the calculation of $\boldsymbol{H_T}(t)$ becomes a weighted linear interpolation between initial and final configuration $\boldsymbol{H_T}(0)$ and $\boldsymbol{H_T}(1)$ with the scale ratio $\gamma$

$$\begin{aligned} \boldsymbol{H_T}(t) &= \gamma(\boldsymbol{I} + t\boldsymbol{H_0}) = \gamma(\boldsymbol{I} + t\ \boldsymbol{H_T}/\gamma - t\ \boldsymbol{I}), \\ &= \gamma(1-t)\ \boldsymbol{I} + t\ \boldsymbol{H_T}. \end{aligned} \tag{31}$$

Unless $\gamma$ is respected, $\boldsymbol{H_T}(t)$ is neither uniform, nor "rigid". The same holds for the interpolation of $\boldsymbol{S}^*$ and $\boldsymbol{s}^*$.

$$\boldsymbol{S}^*(t) = \gamma(1-t)\boldsymbol{S}^*(0) + t\ \boldsymbol{S}^*(1), \tag{32}$$

$$\boldsymbol{s}^*(t) = \gamma(1-t)\boldsymbol{s}^*(0) + t\ \boldsymbol{s}^*(1), \tag{33}$$

The singleton set-points $\boldsymbol{S}^*$ and $\boldsymbol{s}^*$ become a set-point function $\boldsymbol{S}^*(t)$ and $\boldsymbol{s}^*(t)$, the latter in homogeneous image coordinates. The trace of $\boldsymbol{s}^*(t)$ coincides with the image lines connecting initial and target image points. The trajectory of $\boldsymbol{s}^*(t)$, a dynamically varying set of $k$ image points, is however the projection of a uniform, rigid translation. This implies that at any $t$ the current set-point $\boldsymbol{s}^*(t)$ corresponds to an admissible pose of the object, which lies on the shortest path to the target. Moreover, the parameter $t$ corresponds to relative Euclidean distance along the path. Purely image-based interpolation, in contrast, deforms the object and fails to reflect spatial distance.

## 5.4 Velocity profiles

The dynamic formulation now allows to impose any velocity profile on the trajectory. Direct application of $\boldsymbol{S}^*(t) : t \in [0,1]$ yields constant velocity translation parallel to the $z$-axis. Composing $\boldsymbol{S}^*(t)$ with the integral function of a desired "velocity profile" $v(u)$ generates a set-point function of variable gain. Figure 7, for instance, depicts an initial acceleration until saturation, followed by a phase of constant velocity, and a final deceleration until complete set-point overlap $\boldsymbol{S}^*(1)$.

More formally, let $v(u) : u \in [0,1]$ be the desired velocity profile. The resulting time-dilating function $x$ is defined as

$$x(t) = 1/\lambda \int_0^t v(u)du, \text{with } \lambda \text{ s.t. } x(1) = 1. \tag{34}$$

Infact, this procedure corresponds to enrolling the velocity profile by integration, resulting in a one-dimensional trajectory over time, which is finally projected onto the direction of translation.

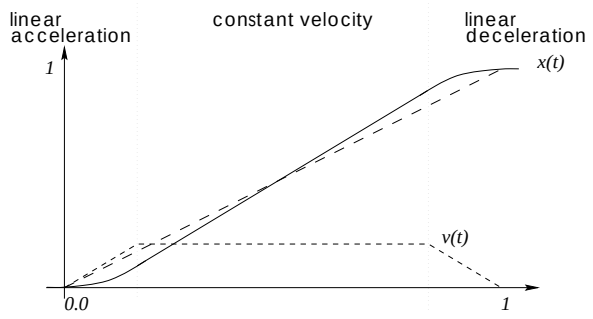


Figure 7: Three-phase velocity profile $v(u)$ and the time-dilation $x(t)$.

# 6 Discussion and Conclusions

In this framework we propose generating task-functions that correspond to complete-path trajectories solely and immediately from the visual information of the scence, without knowledge of the calibration parameters. In classic approaches to task planning, completely known Euclidean structure of the work-space is assumed. Therefore, planning and generation of trajectories are moreless equivalent. In particular, no additional difficulty arises from trajectory generation and from their transfer to the sensor-space.

Unlike these approaches, the presented framework does without Euclidean scene structure, and does without a calibrated sensor. Hence, care has to be taken that the constraints resulting from observing a Euclidean world by a perspective visual sensor are respected by the generated task-functions, or more figuratively, that a link between sensor-space and work-space is kept, that reflects as far as possible the implicit relations between both, as their explicit relationship is not accessible in an uncalibrated environment.

## 6.1 A generated task

We distinguish generating a task-function from planning a task in the sense that no combinatoric search takes place, that no decisions are made. Generating a visual task-function corresponds rather to transforming image information, such that the obtained task-function is a visual instantiation of a generic task, with the current visual configuration of the scene as input parameters (see Fig. 8). The specification of the above mentioned generic tasks is however thought to be in terms of rigid motions of the object, articulated motions of the actuator, or Euclidean geometry of the scene. Thus, at any time the relationship between the visual configuration of features and the geometric and kinematic configuration of tool and actuator have to be identified to the largest extend possible in an uncalibrated setup.

Apart from explicit task planning, the decomposition of observed displacements and composition from observed displacements is included as "construction phase" in the framework of visual trajectory generation. They sometimes require further knowledge about the scene structure. In this report, we have given some consequences and examples of determining the affine properties of the scene. However, the explicit step of affine reconstruction is avoided.

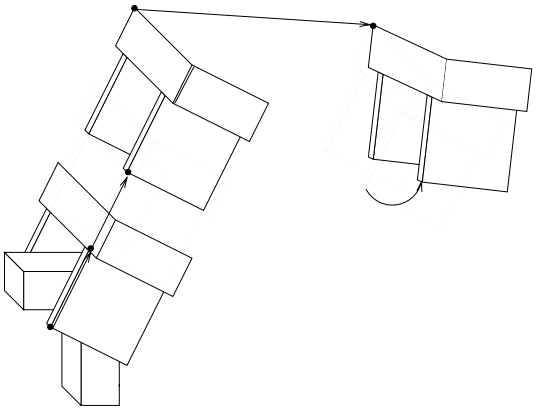

Figure 8: The instantiation of a generic grasping task. Starting from the object to be grasped, the gripper position can be transfered as shown in [5]. Subsequently, a pure translation of twice the claw's edge is instantiated, which avoids an object collision during approaching. Finally, the present position of the gripper is taken and a rotation-first trajectory to reach the initial position of approach is generated.

## 6.2 A plane at infinity

The affine properties of the scene have been proven to be of particular use in the construction and decomposition of projective displacements, the "uncalibrated" representation of rigid spatial motion. The identification of the plane at infinity introduces the concepts of parallelism and oriented directions. They allow to generate the projective translation between an ordered pair of points, and consequently to decompose a general projective displacement into a translation and a rotation around a given center-point. An efficient algebraic method of motion-based affine auto-calibration was given, which nicely fits in the considered system-setup, i.e. controlled robot motions observed by a weakly calibrated stereo-rig.

A further affine property are length ratios. They allow not just for the rigid interpolation of projective motion. Above that, uniformity of the corresponding trajectories in the sense of Euclidean configuration is achieved.

## 6.3 A complete path

It was demonstrated how to inter- and extrapolate projective motion to generate the complete-path of a rigid motion. Here, the path[2] consists primarily of the traces of rigidly moving points. The interpolation of the path is of particular interest since potential local minima in usually point-to-point-wise defined functions of the image error are bridged. An additional contribution to globally convergent control

[2]The ISO defines "path" as "*... spatial locus drawn by the movement of any point on the robot or the workpiece,along which orientation of the end-effector may or may not be variable.*"

is that the path corresponds to motions with monotonically decreasing error in configuration. In conclusion, we consider a complete-path as a step towards globally convergent control.

### 6.4 A uniform trajectory

The difference between a trajectory and a path is that the trajectory describes the progression of motion over time and the object's configuration at a time instant, whereas the path describes the swept trace of motion, only. Thus, the proposed interpolation of projective motion ensures that the reprojected, now dynamically varying set-points correspond at any time to an admissible configuration of the object, i.e. to a rigidly moving object in varying poses. Thus rigidity is preserved, trajectories are physically valid.

Over and above that, the interpolation parameter $t$ establishes an implicit relation between visual and Euclidean configuration, and the corresponding error functions. In other words, the visual set-points at $t$ correspond to the $t$-fractile of distance in Euclidean configuration. One can also view this relationship as an implicit calibration of and only of the currently instantiated visual task with respect to (relativ) error in configuration.

We consider the established relationship between visual and Euclidean configuration as a step towards weaker requirements on hand-eye and kinematic actuator calibration [7], [8]. Furthermore, there is now an apparent connection between the gain in visual configuration and the gain in object configuration, which allows for optimizing with respect to dynamic properties, e.g. maximal acceleration, speed limits of the actuator [9].

## References


[1] F. Devernay and O. Faugeras. From projective to euclidean reconstruction. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, San Francisco, California, USA*, June 1996.

[2] B. Espiau. Sur les erreurs en asservissement visuel. Rapport de recherche, INRIA, July 1995.

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

[4] O. Faugeras. What can be seen in three dimensions with an uncalibrated stereo rig? In G. Sandini, editor, *Proceedings of the 2nd European Conference on Computer Vision, Santa Margherita Ligure, Italy*, pages 563–578. Springer-Verlag, May 1992.

[5] R. Horaud, F. Dornaika, and B. Espiau. Visually guided object grasping. IEEE *Transactions on Robotics and Automation*, 1997. accepted.

[6] S. Hutchinson, G. D. Hager, and P. I. Corke. A tutorial on visual servo control. *IEEE Transactions on Robotics and Automation*, 12(5):651–669, October 1996.

[7] M. Jaegersand. On-line estimation of visual-motor models using active vision. In *Proceedings of the ARPA Image Understanding Workshop*, 1996.

[8] M. Jaegersand. Visual servoing using trust region methods and estimation of the full coupled visual-motor jacobian. In *Proceedings of IASTED Applications of Control and Robotics*, pages 105–108, 1996.

[9] E. Marchand, A. Rizzo, and F. Chaumette. Avoiding robot joint limits and kinematic singularities in visual servoing. In *Proceedings of 13th IARP/IEEE International Conference in Pattern Recognition*, volume A, pages 297 – 301, 1996.

[10] C. Samson, B. Espiau, and M. Le Borgne. *Robot Control: The Task Function Approach*. Clarendon Press, Oxford, 1991.

[11] B. Triggs. Factorization methods for projective structure and motion. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, San Francisco, California, USA*, pages 845–851, 1996.

[12] L.E. Weiss, A.C. Anderson, and C.P. Neuman. Dynamic sensor-based control of robots with visual feedback. IEEE *Transactions on Pattern Analysis and Machine Intelligence*, 3(5):404–417, October 1987.

[13] A. Zisserman, P.A. Beardsley, and I.D. Reid. Metric calibration of a stereo rig. In *Workshop on Representation of Visual Scenes, Cambridge, Massachusetts, USA*, pages 93–100, June 1995.