# SIX DEGREE-OF-FREEDOM TASK ENCODING IN VISION-BASED CONTROL SYSTEMS

**Wen-Chung Chang** [*,1] **A. S. Morse** [**,2]

[*] *Institute of Electrical Engineering, National Dong Hwa University, Shoufeng, Hualien 974, Taiwan, R.O.C.*
*chang@control.ee.ndhu.edu.tw*
[**] *Department of Electrical Engineering, Yale University, P.O.Box 208267, New Haven, CT 06520-8267.*
*morse@sysc.eng.yale.edu*

Abstract: Issues concerning task encoding in vision-based control systems have recently been discussed in (Chang *et al.*, 1997). It is shown that in the absence of measurement noise *precise* positioning sometimes can possibly be achieved despite camera model imprecision. The purpose of this paper is to extend the analysis to the general six degree-of-freedom problems.

Keywords: Robots; Tasks; Encoding; Set-point control; Stereo vision.

## 1. INTRODUCTION

Feedback control systems employing video cameras as sensors have been studied in the robotics community for many years {c.f (Hutchinson *et al.*, 1996)}. An especially interesting feature of such systems is that *both* the process state and the reference set-point are typically observed through the same sensors {i.e., cameras}. Because of this unusual architectural feature, it is sometimes possible to achieve *precise* positioning {in the absence of measurement noise}, despite sensor/actuator and process model imprecision, just as it is in the case of a conventional set-point control system with a loop-integrator and fixed exogenous reference. But in contrast to a set-point control system where what to choose for an error is usually clear, in vision-based systems there are many choices for errors, each with different attributes. The aim of this paper is to discuss these issues in a fairly general setting and to provide concrete examples to illustrate the concepts involved in geometrical terms. The analysis presented in this paper is for the general six degree-of-freedom problems which is extended from the result for three degree-of-freedom motion in (Chang *et al.*, 1997).

## 2. THE SYSTEM

This paper considers the problem of positioning a rigid robot in a prescribed *workspace* $\mathcal{X} \subset SE^3$ [‡] using data observed by a two-camera vision system. We denote by $\mathcal{A} \subset \mathbb{R}^3$ the set of possible "positions" of the origin of the frame rigidly attached to the robot which we henceforth call the *robot frame*. Meanwhile, $\mathcal{B} \subset SO^3$ [†] defines the set of possible "orientations" of the robot frame. The workspace $\mathcal{X} \subset SE^3$ which defines all possible positions and orientations that the robot can attain can be written as $\mathcal{X} \triangleq \mathcal{A} \times \mathcal{B}$.

The observed data consists of various geometrical features of the robot and the environment. These

[1] Chang was supported by the National Science Council, R.O.C., under grant NSC 88-2213-E-259-006.
[2] Morse was supported by the NSF, AFOSR, and ARO.

[‡] $SE^3$ stands for Special Euclidean group of order 3.
[†] $SO^3$ stands for Special Orthogonal group of order 3.

geometrical features of interest appear in the *two-camera field of view* $\mathcal{V} \subset \mathbb{R}^3$. Invariably $\mathcal{A} \subset \mathcal{V}$. Let the *viewable workspace* $\mathcal{W}$ define all possible positions and orientations that the robot can attain if $\mathcal{A}$ were equal to $\mathcal{V}$, i.e., $\mathcal{W} \triangleq \mathcal{V} \times \mathcal{B}$.

For each *robot pose* $x \in \mathcal{X}$, $x$ can be written as $x = \{r, R\}$ where $r \in \mathcal{A}$ and $R \in \mathcal{B}$. The robot is assumed to admit a simple kinematic model (Spong and Vidyasagar, 1989) of the form

$$\dot{r} = v \ , \ \dot{R} = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_1 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} R$$

$$u \triangleq \begin{bmatrix} v \\ \omega \end{bmatrix} \ , \ \omega \triangleq \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix}$$

where $u$ is a control vector composed of the translational and angular velocities of the robot frame with respect to a fixed frame each being a vector in $\mathbb{R}^3$.

The robot pose and the *target pose* {the desired robot pose} in $\mathcal{X}$ are determined by various geometrical features of interest in $\mathcal{V}$ which are seen in an *image space* $\mathcal{Y} \triangleq \mathbb{R}^2 \oplus \mathbb{R}^2$ through a fixed but unknown *two-camera model* $G : \mathcal{V} \to \mathcal{Y}$ which describes a two-camera vision system. Thus, a specific feature $\phi$ in $\mathcal{V}$ is seen in $\mathcal{Y}$ as [3]

$$\zeta = G(\phi). \tag{1}$$

Note that because $\zeta$ is a subset of $\mathcal{Y}$, it is also an element of the power set [4] of $\mathcal{Y}$ which we henceforth denote by $\bar{\mathcal{Y}}$. Similarly $\phi$ is an element of the power set of $\mathcal{V}$, which we write as $\bar{\mathcal{V}}$.

Since the state $x$ is not measurable whereas the $G$-images of various geometrical features which determine the robot pose and the target pose are, in order for the robot/vision system to be observable one would need $G$ to be at least injective. Thus, one could design a control law $u$ to accomplish a positioning task. On the other hand, one would never expect to know precisely what $G$ is. Accordingly, we will assume that $G$ is a fixed but unknown element of a prescribed set $\mathcal{G}$ of injective functions each mapping $\mathcal{V}$ into $\mathcal{Y}$. It will be useful to index $\mathcal{G}$ by a parameter $p$ taking values in some appropriate index set $\mathcal{P}$. That is $\mathcal{G} = \{G_p : \ p \in \mathcal{P}\}$ and $G = G_\omega$ where $\omega$ is some fixed but unknown element of $\mathcal{P}$.

For each two-camera model $G_p \in \mathcal{G}$, there is a naturally induced injective function $\bar{G}_p$ from $\bar{\mathcal{V}}$ to $\bar{\mathcal{Y}}$ defined by the rule $\bar{x} \mapsto G_p(\bar{x})$. In the sequel $\bar{G} \triangleq \bar{G}_\omega$. In this notation (1) can also be written as $\zeta = \bar{G}(\phi)$, but (1) is preferred.

[3] With $\mathcal{A}$ and $\mathcal{B}$ sets, $\mathcal{C} \subset \mathcal{A}$, and $f : \mathcal{A} \to \mathcal{B}$, we write $f(\mathcal{C})$ for the $f$-image of $\mathcal{C}$ which is defined by $\{f(a) \ : \ a \in \mathcal{C}\}$.

[4] The power set of $\mathcal{A}$ is the set of all subsets of $\mathcal{A}$.

## 3. THE TASK

It is assumed that the robot pose $x$ can be uniquely determined by an ordered set of $m$ observed robot features $\{\phi_{x_1}, \phi_{x_2}, \ldots, \phi_{x_m}\}$ where each $\phi_{x_i}$ is a member of some specific class $\mathcal{F}_{x_i}$ of like or similar objects on the robot. For example, $\mathcal{F}_{x_1}$ might be a family of sets of a line segment, $\mathcal{F}_{x_2}$ a family of sets of a single point, etc. In the sequel $\mathcal{F}_{x_1}, \mathcal{F}_{x_2}, \ldots, \mathcal{F}_{x_m}$ are $m$ such feature classes and $\mathcal{F}_x$ is a given *composite robot feature space* consisting of all possible lists $\{\phi_{x_1}, \phi_{x_2}, \ldots, \phi_{x_m}\}$ of robot features, where each $\phi_{x_i} \in \mathcal{F}_{x_i}$. $\mathcal{F}_x$ is a prespecified subset of $\mathcal{F}_{x_1} \times \mathcal{F}_{x_2} \times \cdots \times \mathcal{F}_{x_m}$ which is in turn a subset of $\widetilde{\mathcal{V}}_x \triangleq \underbrace{\bar{\mathcal{V}} \times \bar{\mathcal{V}} \times \cdots \times \bar{\mathcal{V}}}_{m \text{ times}}$. Thus, the robot pose can be determined by a known function $H_x : \mathcal{F}_x \to \bar{\mathcal{W}}$, typically *injective*, i.e.,

$$\{x\} = H_x(f_x).$$

Similarly, target pose is assumed to be determined by an ordered set of $n$ simultaneously observed target features $\{\phi_{d_1}, \phi_{d_2}, \ldots, \phi_{d_n}\}$ where each $\phi_{d_i}$ is a member of some specific class $\mathcal{F}_{d_i}$ of like or similar objects. In the sequel $\mathcal{F}_{d_1}, \mathcal{F}_{d_2}, \ldots, \mathcal{F}_{d_n}$ are $n$ such feature classes and $\mathcal{F}_d$ is a given *composite target feature space* consisting of all lists $\{\phi_{d_1}, \phi_{d_2}, \ldots, \phi_{d_n}\}$ of interest, where each $\phi_{d_i} \in \mathcal{F}_{d_i}$. Thus $\mathcal{F}_d$ is a prespecified subset of $\mathcal{F}_{d_1} \times \mathcal{F}_{d_2} \times \cdots \times \mathcal{F}_{d_n}$ which is in turn a subset of $\widetilde{\mathcal{V}}_d \triangleq \underbrace{\bar{\mathcal{V}} \times \bar{\mathcal{V}} \times \cdots \times \bar{\mathcal{V}}}_{n \text{ times}}$. Thus, the target pose $x_d \in \mathcal{X}$ can be determined by a known function $H_d : \mathcal{F}_d \to \bar{\mathcal{W}}$, i.e.,

$$x_d \in H_d(f_d).$$

Note that, $H_d$ in general is not injective. Hence, the *task* to be accomplished is

$$\boxed{H_x(f_x) \subset H_d(f_d).} \tag{2}$$

When one wishes to define the same task in composite feature space, one has to look into the two functions $H_x$ and $H_d$. That is, one needs to consider a *unified composite feature space* where the task (2) can be easily defined by set-equality {or set-inclusion in more general situations}. We assume that the robot pose can also be computed based on an ordered set of $l$ {may not be the same as $m$} geometrical features of the robot $\{\phi_1, \phi_2, \ldots, \phi_l\}$ which we call a *unified list of robot features* and is in turn determined by a list of "observed" robot features. Each $\phi_i$ in a unified list of robot features is a member of a specific class $\mathcal{F}_i$ of similar geometrical features of a frame. In the sequel, $\mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_l$ are $l$ such feature classes and $\mathcal{F}$ is a given *unified composite feature space* consisting of lists $\{\phi_1, \phi_2, \ldots, \phi_l\}$ of geometrical features of interest in $\mathcal{V}$, where each $\phi_i \in \mathcal{F}_i$. Thus $\mathcal{F}$ is a prespecified subset of $\mathcal{F}_1 \times \mathcal{F}_2 \times \cdots \times \mathcal{F}_l$ which is in turn a subset of $\widetilde{\mathcal{V}} \triangleq \underbrace{\bar{\mathcal{V}} \times \bar{\mathcal{V}} \times \cdots \times \bar{\mathcal{V}}}_{l \text{ times}}$. Furthermore, we assume that there exist two *injective* maps one from $\mathcal{F}_x$ to $\mathcal{F}$ and the other from $\mathcal{F}$ to $\bar{\mathcal{W}}$. Specifically, one needs to define two injective functions $H : \mathcal{F} \to \bar{\mathcal{W}}$ referred to as *pose map* and $T_x : \mathcal{F}_x \to \mathcal{F}$ called a *unified robot feature map* such that the following equality holds.

$$H_x = H \circ T_x$$

As in the case of computing the robot pose, the target pose can also be computed based on an ordered set of $l$ geometrical features of the target in $\mathcal{V}$ which we call a *unified list of target features* and is in turn determined by a list of observed target features. Thus, one needs to define a function $T_d : \mathcal{F}_d \to \mathcal{F}$ called a *unified target feature map* such that the following equality holds.

$$H_d = H \circ T_d$$

Therefore, the relationship between observed lists of robot features and unified lists of robot features is characterized by a given *unified robot feature map* $T_x : \mathcal{F}_x \to \mathcal{F}$ which maps $f_x$ in the composite robot feature space to $T_x(f_x)$ in the unified composite feature space.

Let $\mathcal{D} \subset \mathcal{F}$ denote the unified composite feature set consisting of all lists of geometrical features of interest in $\mathcal{A}$, i.e., $\mathcal{D}$ is a prespecified subset of $(\mathcal{F}_1 \cap \bar{\mathcal{A}}) \times (\mathcal{F}_2 \cap \bar{\mathcal{A}}) \times \cdots \times (\mathcal{F}_l \cap \bar{\mathcal{A}})$ which is in turn a subset of $\widetilde{\mathcal{A}} \triangleq \underbrace{\bar{\mathcal{A}} \times \bar{\mathcal{A}} \times \cdots \times \bar{\mathcal{A}}}_{l \text{ times}}$. Each observed list of robot features $f_x$ specifies a unified list of robot features $T_x(f_x) \in \mathcal{D}$ which in turn uniquely determines the robot pose via a fixed and known injective *pose map* $H : \mathcal{F} \to \bar{\mathcal{W}}$, i.e.,

$$\{x\} = [H \circ T_x](f_x).$$

Meanwhile, the relationship between observed lists of target features and unified lists of target features is characterized by a given *unified target feature map* $T_d : \mathcal{F}_d \to \mathcal{F}$ which maps $f_d$ in the composite target feature space to $T_d(f_d) \in \mathcal{D}$ in the unified composite feature space such that $[H \circ T_d](f_d) \subset \mathcal{X}$. Thus the target pose $x_d \in \mathcal{X}$ can be defined by the formula

$$x_d \in [H \circ T_d](f_d).$$

Hence, the *task* (2) is equivalent to

$$[H \circ T_x](f_x) \subset [H \circ T_d](f_d). \tag{3}$$

Note that by virtue of the injectivity of $H$, the *task* (3) thus can be restated as

$$\boxed{T_x(f_x) \subset T_d(f_d).} \tag{4}$$

## 4. ENCODING

### 4.1 *Cartesian-Based Approach*

The widely used Cartesian-based approach (Wilson *et al.*, 1996) is motivated by the heuristic idea of "certainty equivalence". In the present context, certainty equivalence advocates that one should use estimates of $f_x$ and $f_d$ to accomplish task (4), with the understanding that these estimates are to be viewed as correct even though they may not be. The construction of such estimates starts with the selection {by some means} of a two-camera model $G_q$ in $\mathcal{G}$ which, in the context of certainty equivalence, is considered to be an approximate model of $G$.

To develop an estimate of $f_x$, we need first to get a compact expression relating $f_x$ to what can be observed, namely the list $\{G(\phi_{x_1}), G(\phi_{x_2}), \ldots, G(\phi_{x_m})\}$. Thus, for each $p \in \mathcal{P}$, define the map $\widetilde{G}_{x_p} : \mathcal{F}_x \to \widetilde{\mathcal{Y}}_x \triangleq \underbrace{\bar{\mathcal{Y}} \times \bar{\mathcal{Y}} \times \cdots \times \bar{\mathcal{Y}}}_{m \text{ times}}$ by the rule

$$\{\phi_{x_1}, \phi_{x_2}, \ldots, \phi_{x_m}\} \mapsto \{G_p(\phi_{x_1}), G_p(\phi_{x_2}), \ldots, G_p(\phi_{x_m})\}$$

and write $\widetilde{G}_x$ for $\widetilde{G}_{x_\omega}$. The preceding notation enables us to write compactly

$$y = \widetilde{G}_x(f_x) \tag{5}$$

where $y$ is the measured list of observed robot features $y \triangleq \{G(\phi_{x_1}), G(\phi_{x_2}), \ldots, G(\phi_{x_m})\}$. For purposes of certainty equivalence $G_q$ is our current estimate of $G$, therefore it makes sense to get an estimate of $f_x$ using a left inverse of $\widetilde{G}_{x_q}$. Such left inverses exist because the $\widetilde{G}_{x_p}$ are injective. This, in turn, is a direct consequence of the assumed injectivity of the $G_p$. In view of (5), it is natural to define

$$\widehat{f}_x \triangleq \widetilde{G}_{x_q}^{-1}(y)$$

as the estimate of $f_x$ to be considered.

Similarly, to develop a corresponding estimate of $f_d$, define, for each $p \in \mathcal{P}$, the map $\widetilde{G}_{d_p} : \mathcal{F}_d \to \widetilde{\mathcal{Y}}_d \triangleq \underbrace{\bar{\mathcal{Y}} \times \bar{\mathcal{Y}} \times \cdots \times \bar{\mathcal{Y}}}_{n \text{ times}}$ by the rule

$$\{\phi_{d_1}, \phi_{d_2}, \ldots, \phi_{d_n}\} \mapsto \{G_p(\phi_{d_1}), G_p(\phi_{d_2}), \ldots, G_p(\phi_{d_n})\}$$

and write $\widetilde{G}_d$ for $\widetilde{G}_{d_\omega}$. We thus can write

$$z = \widetilde{G}_d(f_d)$$

where $z$ is the measured list of observed features $z \triangleq \{G(\phi_{d_1}), G(\phi_{d_2}), \ldots, G(\phi_{d_n})\}$. Therefore, one can get an estimate of $f_d$ using a left inverse of $\widetilde{G}_{d_q}$, i.e.,

$$\widehat{f}_d \triangleq \widetilde{G}_{d_q}^{-1}(z).$$

Since $\widetilde{G}_{x_q}^{-1}(y)$ and $\widetilde{G}_{d_q}^{-1}(z)$ are estimates of $f_x$ and $f_d$ respectively, in accordance with certainty equivalence, to achieve the task (4) one should seek a control $u$ to achieve the encoded task

$$\boxed{\left[T_x \circ \widetilde{G}_{x_q}^{-1}\right](y) \subset \left[T_d \circ \widetilde{G}_{d_q}^{-1}\right](z).} \tag{6}$$

When does achieving the encoded task (6) imply that the original task (4) has been achieved as well? The following lemma gives a partial answer.

*Lemma 4.1.* Let $q \in \mathcal{P}$ be given and suppose that $\widetilde{G}_q^{-1} : \bar{\mathcal{Y}}^l \to \mathcal{F}$ is a fixed left inverse of $\widetilde{G}_q : \mathcal{F} \to \widetilde{\mathcal{Y}} \stackrel{\triangle}{=} \underbrace{\bar{\mathcal{Y}} \times \bar{\mathcal{Y}} \times \cdots \times \bar{\mathcal{Y}}}_{l \text{ times}}$ which is defined by the rule

$$\{\phi_1, \phi_2, \ldots, \phi_l\} \mapsto \{G_q(\phi_1), G_q(\phi_2), \ldots, G_q(\phi_l)\}$$

and write $\widetilde{G}$ for $\widetilde{G}_\omega$. For each $f_x \in \mathcal{F}_x$ and each $f_d \in \mathcal{F}_d$ such that both $T_x(f_x)$ and $T_d(f_d)$ are in $\mathcal{D}$, (6) implies (4) provided

$$\widetilde{G}_q^{-1} \circ \widetilde{G}$$

is injective on $\mathcal{D}$ and

$$\left[\widetilde{G}_q^{-1} \circ \widetilde{G} \circ T_x\right](f_x) = \left[T_x \circ \widetilde{G}_{x_q}^{-1} \circ \widetilde{G}_x\right](f_x) \tag{7}$$

$$\left[\widetilde{G}_q^{-1} \circ \widetilde{G} \circ T_d\right](f_d) = \left[T_d \circ \widetilde{G}_{d_q}^{-1} \circ \widetilde{G}_d\right](f_d). \tag{8}$$

In the sequel, we say that any pair $\{\widetilde{G}_p, \widetilde{G}_p^{-1}\}$, with $p \in \mathcal{P}$ and $\widetilde{G}_p^{-1}$ a left inverse of $\widetilde{G}_p$, is an *admissible bi-model* if [5]

$$\left[\widetilde{G}_p^{-1} \circ \widetilde{G}\right] \Big| \mathcal{D}$$

is an injective function. Note that because $\left[\widetilde{G}^{-1} \circ \widetilde{G}\right] \Big| \mathcal{D}$ is the identity on $\mathcal{D}$ and therefore injective, $\{\widetilde{G},\ \widetilde{G}^{-1}\}$ is an admissible bi-model. As regards robustness, this means that under suitable technical conditions, if $\mathcal{G}$ is a sufficiently small open neighborhood about $G$, then each $G_p \in \mathcal{G}$ would have a left inverse of $\widetilde{G}_p$ which makes $\{\widetilde{G}_p,\ \widetilde{G}_p^{-1}\}$ an admissible bi-model.

In most applications of interest, one would like to be able to achieve (4) no matter what list of robot features $f_x \in \mathcal{F}_x$ and list of target features $f_d \in \mathcal{F}_d$ might be. One way to insure that this is possible, is to require that (7) hold for all $f_x \in \mathcal{F}_x$ and (8) hold for all $f_d \in \mathcal{F}_d$; i.e.,

$$\widetilde{G}_q^{-1} \circ \widetilde{G} \circ T_x = T_x \circ \widetilde{G}_{x_q}^{-1} \circ \widetilde{G}_x \tag{9}$$

$$\widetilde{G}_q^{-1} \circ \widetilde{G} \circ T_d = T_d \circ \widetilde{G}_{d_q}^{-1} \circ \widetilde{G}_d. \tag{10}$$

When (9) holds we say that $\widetilde{G}_q^{-1} \circ \widetilde{G}$ and $T_x$ *commute.* Meanwhile, when (10) holds we say that $\widetilde{G}_q^{-1} \circ \widetilde{G}$ and $T_d$ *commute.* There is of course no reason to expect that these commuting properties will hold; typically they do not unless, in the case of pure translation {i.e.,$\mathcal{X} \stackrel{\triangle}{=} \mathcal{A}$}. One way to deal with this problem is discussed next.

### 4.2 *Modified Cartesian-Based Approach*

The modified Cartesian-based approach, which seems to be new, is motivated by a desire to avoid the stringent commuting requirements (9) and (10). The starting point for the approach is the requirements that both the unified robot feature map $T_x$ and the unified target feature map $T_d$ be "invariant" on $\mathcal{G}$. Invariance is defined as follows.

A unified robot feature map $T_x$ is said to be *invariant* on $\mathcal{G}$ if there exists a function $\widetilde{T}_x : \widetilde{\mathcal{Y}}_x \to \widetilde{\mathcal{Y}}$ such that

$$\widetilde{G}_p \circ T_x = \widetilde{T}_x \circ \widetilde{G}_{x_p}, \qquad p \in \mathcal{P}. \tag{11}$$

Similarly, a unified target feature map $T_x$ is said to be *invariant* on $\mathcal{G}$ if there exists a function $\widetilde{T}_d : \widetilde{\mathcal{Y}}_d \to \widetilde{\mathcal{Y}}$ such that

$$\widetilde{G}_p \circ T_d = \widetilde{T}_d \circ \widetilde{G}_{d_p}, \qquad p \in \mathcal{P}. \tag{12}$$

$\widetilde{T}_x$ and $\widetilde{T}_d$ can be seen as functions that map the two-camera image of a list of features into the two-camera image of the corresponding unified feature sets. As we will see, for example for perspective projection camera models, demanding that the invariance properties hold turns out to be much less severe than requiring the commuting properties (9) and (10) to hold {c.f. Section 5}.

To proceed, assume that two functions $\widetilde{T}_x$ and $\widetilde{T}_d$ have been found for which the invariance properties hold. As in the Cartesian-based approach, to encode one also needs to select a $G_q$ in $\mathcal{G}$ which in the context of certainty equivalence, is considered to be an estimate of $G$. We assume that such a $G_q$ has been chosen and that $\widetilde{G}_q^{-1}$ is a fixed left inverse of $\widetilde{G}_q$. In contrast to the Cartesian-based approach which seeks to accomplish the task defined by (6), the modified Cartesian approach seeks to achieve

$$\boxed{\left[\widetilde{G}_q^{-1} \circ \widetilde{T}_x\right](y) \subset \left[\widetilde{G}_q^{-1} \circ \widetilde{T}_d\right](z)} \tag{13}$$

instead. The following lemma provides justification for this approach.

*Lemma 4.2.* [†] Let $q \in \mathcal{P}$ be given and suppose that $\widetilde{G}_q^{-1} : \bar{\mathcal{Y}}^l \to \mathcal{F}$ is a fixed left inverse of

[5] With $\mathcal{A}$ and $\mathcal{B}$ sets, $\mathcal{C} \subset \mathcal{A}$, and $f : \mathcal{A} \to \mathcal{B}$, we write $f|\mathcal{C}$ for the *restricted* map $\mathcal{C} \to f(\mathcal{C}) : c \mapsto f(c)$.

[†] It should be emphasized that lemma 4.2 does not require $G_q$ to be "close" to $G$ in any particular sense.

$\widetilde{G}_q : \mathcal{F} \to \widetilde{\mathcal{Y}} \triangleq \underbrace{\bar{\mathcal{Y}} \times \bar{\mathcal{Y}} \times \cdots \times \bar{\mathcal{Y}}}_{l \text{ times}}$ which is defined by the rule

$$\{\phi_1, \phi_2, \ldots, \phi_l\} \mapsto \{G_q(\phi_1), G_q(\phi_2), \ldots, G_q(\phi_l)\}$$

and write $\widetilde{G}$ for $\widetilde{G}_\omega$. Suppose in addition that $\widetilde{T}_x$ and $\widetilde{T}_d$ are two fixed functions for which the invariance properties (11) and (12) hold. For each $f_x \in \mathcal{F}_x$ and each $f_d \in \mathcal{F}_d$ such that both $T_x(f_x)$ and $T_d(f_d)$ are in $\mathcal{D}$, (13) implies (4) provided $\{\widetilde{G}_q, \widetilde{G}_q^{-1}\}$ is an admissible bi-model.

### 4.3 *Image-Based Approach*

As with the modified Cartesian-based approach, the image-based approach (Hager *et al.*, 1995) also requires that both unified robot feature map $T_x$ and unified target feature map $T_d$ be invariant on $\mathcal{G}$. To proceed, assume that the invariance properties are satisfied by some computable functions $\widetilde{T}_x$ and $\widetilde{T}_d$. In contrast to the modified Cartesian approach, the image-based approach seeks to achieve

$$\boxed{\widetilde{T}_x(y) \subset \widetilde{T}_d(z).} \tag{14}$$

The approach is justified by the following lemma.

*Lemma 4.3.* Suppose that $\widetilde{T}_x$ and $\widetilde{T}_d$ are fixed functions for which the invariance properties (11) and (12) hold respectively. For each $f_x \in \mathcal{F}_x$ and each $f_d \in \mathcal{F}_d$, (14) implies (4).

Achieving the task defined by (14) clearly causes the task defined by (13) to be achieved. But, as opposed to both Cartesian-based approaches, the image-based approach does not require the selection of a candidate two-camera model $G_p$ in $\mathcal{G}$ to serve as an estimate of $G$. However, in practice, designing a controller that achieves (14) may require some estimate of $G$.

## 5. EXAMPLES

We assume that a family of admissible two-camera models $\mathcal{G}$ is given such that each $G_p$ in $\mathcal{G}$ is of the form of *perspective projection* model (Horn, 1986). In the sequel, $L(p_1, p_2)$ defines the line passing through the two points $p_1$ and $p_2$.

$\Gamma_1(x_1, x_2, x_3, x_4) \triangleq x_1 + \alpha_1(x_3 - x_1)$, where

$x_{i(i:1,2,3,4)} \in \mathbb{R}^3$ and $\alpha_1 \in \mathbb{R}$ such that

$x_1 + \alpha_1(x_3 - x_1) = x_2 + \alpha_2(x_4 - x_2)$ for some $\alpha_2 \in \mathbb{R}$.

$\Gamma_2(y_1, y_2, y_3, y_4) \triangleq y_1 + \mathrm{diag}\{\beta_1, \beta_1, \beta_3, \beta_3\}(y_3 - y_1)$, where

$y_{i(i:1,2,3,4)} \in \mathbb{R}^4$ and $\beta_1, \beta_3 \in \mathbb{R}$ such that

$y_1 + \mathrm{diag}\{\beta_1, \beta_1, \beta_3, \beta_3\}(y_3 - y_1) = y_2$
$+\mathrm{diag}\{\beta_2, \beta_2, \beta_4, \beta_4\}(y_4 - y_2)$ for some $\beta_2, \beta_4 \in \mathbb{R}$.

*Example 5.1.* (6-DOF Point-to-Point Positioning). The task is to drive the robot pose $x$ determined by three observed feature points $\{x_1, x_2, x_3\}$ to a set-point $s$ in the workspace $\mathcal{X}$ which is determined by three observed feature points $\{s_1, s_2, s_3\}$ as shown in Fig. 1. Note that it is assumed that the geometric relation between the three observed feature points of the robot is the same as the geometric relation between the three observed feature points of the target, i.e., there exists a function $\varpi$ such that $\varpi(x_1, x_2, x_3) = 0$ and $\varpi(s_1, s_2, s_3) = 0$. The three feature points

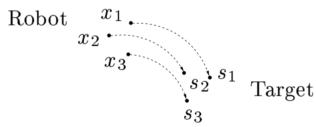


Fig. 1. 6-DOF Point-to-Point Positioning

of the robot $x_1, x_2, x_3 \in \mathcal{A}$ and the three feature points of the target $s_1, s_2, s_3 \in \mathcal{A}$ are sensed by the cameras. In this example, $m$, $n$, and $l$ are all equal to 3. The composite robot feature space is the same as the composite target feature space and is the family of 3 ordered sets each with a single element in $\mathcal{A}$ and these 3 elements satisfy the geometric relation defined by $\varpi$, i.e., $\mathcal{F}_x = \mathcal{F}_d = \Big\{\{\{s_1\}, \{s_2\}, \{s_3\}\} \,|s_i \in \mathcal{A}, \varpi(s_1, s_2, s_3) = 0\Big\}$. Thus, the unified composite feature space is defined to be the same space. Hence, the unified robot feature map $T_x$ and the unified target feature map are both identity maps. Obviously, these maps are invariant on $\mathcal{G}$ since one can take both $\widetilde{T}_x$ and $\widetilde{T}_d$ to be the identity on $\widetilde{\mathcal{Y}}$ (c.f. (11) and (12)). Moreover, for any admissible bi-model $\{\widetilde{G}_p, \widetilde{G}_p^{-1}\}$, it is also true that $T_x$ and $G_p^{-1} \circ G$ commute (c.f. (9)) and $T_d$ and $G_p^{-1} \circ G$ commute (c.f. (10)). Therefore, all three approaches assure precise positioning.

*Example 5.2.* (6-DOF Point-to-Point Positioning with Multiple Features). The task is to drive the robot pose $x$ which is determined by four observed coplanar feature points $\{x_{i(i:1,2,3,4)}\}$ to the target pose $x_d \in \mathcal{X}$ which is determined by eight observed coplanar feature points $\{s_{i(i:1,2,\cdots,8)}\}$ as shown in Fig. 2. The robot pose can be determined by two observed feature points of the robot and one intersecting point defined by the four observed feature points of the robot. Meanwhile, the target pose can be determined by three intersecting points defined by the eight observed feature points of the target. Note that it is assumed that these two ordered sets of three points defined by observed feature points of the robot and the target respectively are well defined such that they have the same geometric relation described by a function $\varpi$. The four feature points of the robot $x_{i(i:1,2,3,4)} \in \mathcal{A}$ and the eight feature points of the target $s_{i(i:1,2,\cdots,8)} \in \mathcal{V}$ are sensed by the cameras.

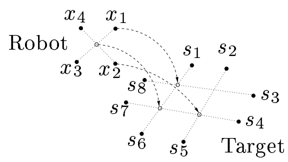


Fig. 2. 6-DOF Point-to-Point Positioning with Multiple Features

Thus, $m$ is equal to 4 and $n$ is equal to 8. The composite robot feature space $\mathcal{F}_x$ is defined by

$$\mathcal{F}_x = \Big\{ \{\{x_1\},\{x_2\},\{x_3\},\{x_4\}\} \,\big|\, x_{i(i:1,2,\cdots,5)} \in \mathcal{A} \text{ and } L(x_1,x_3) \cap L(x_2,x_4) = \{x_5\} \text{ such that } \varpi(x_1,x_2,x_5) = 0 \Big\}.$$

Meanwhile, the composite target feature space $\mathcal{F}_d$ is defined by

$$\mathcal{F}_d = \Big\{ \{\{s_i\}_{(i:1,2,\cdots,8)}\} \,\big|\, s_{i(i:1,2,\cdots,8)} \in \mathcal{V}, s_{i(i=9,10,11)} \in \mathcal{A}, L(s_1,s_6) \cap L(s_3,s_8) = \{s_9\}, L(s_2,s_5) \cap L(s_4,s_7) = \{s_{10}\}, L(s_1,s_6) \cap L(s_4,s_7) = \{s_{11}\}, \text{ and such that } \varpi(s_9,s_{10},s_{11}) = 0 \Big\}.$$

In this example, $l$ is equal to 3. The unified composite feature space $\mathcal{F}$ is defined by

$$\mathcal{F} = \Big\{ \{\{w_1\},\{w_2\},\{w_3\}\} \,\big|\, w_1,w_2,w_3 \in \mathcal{A} \text{ and such that } \varpi(w_1,w_2,w_3) = 0 \Big\}.$$

The unified robot feature map $T_x : \mathcal{F}_x \to \mathcal{F}$ is defined by the rule

$$\{\{x_1\},\{x_2\},\{x_3\},\{x_4\}\} \mapsto \{\{x_1\},\{x_2\},\{\Gamma_1(x_1,x_2,x_3,x_4)\}^{\dagger}\}.$$

Meanwhile, the unified target feature map $T_d : \mathcal{F}_d \to \bar{\mathcal{F}}$ is defined by the rule

$$\{\{s_i\}_{(i:1,2,\cdots,8)}\} \mapsto \{\{\Gamma_1(s_1,s_3,s_6,s_8)\}^{\dagger}, \{\Gamma_1(s_2,s_4,s_5,s_7)\}^{\dagger}, \{\Gamma_1(s_1,s_4,s_6,s_7)\}^{\dagger}\}.$$

Defining $\widetilde{T}_x : \widetilde{\mathcal{Y}}_x \to \widetilde{\mathcal{Y}}$ by the rule

$$\{\{g_1\},\{g_2\},\{g_3\},\{g_4\}\} \mapsto \{\{g_1\},\{g_2\},\{\Gamma_2(g_1,g_2,g_3,g_4)\}\},$$

which maps the four points $g_1,g_2,g_3,g_4 \in \mathcal{Y}$ to three points in $\mathcal{Y}$. The first two points are $g_1$ and $g_2$. And, the third point is the intersection of two lines defined by $\{g_1,g_3\}$ and $\{g_2,g_4\}$. Meanwhile, defining $\widetilde{T}_d : \widetilde{\mathcal{Y}}_d \to \mathcal{Y}$ by the rule

$$\{\{h_i\}_{(i:1,2,\cdots,8)}\} \mapsto \{\{\Gamma_2(h_1,h_3,h_6,h_8)\}, \{\Gamma_2(h_2,h_4,h_5,h_7)\}, \{\Gamma_2(h_1,h_4,h_6,h_7)\}\},$$

which maps the eight points $h_{i(i:1,2,\cdots,8)} \in \mathcal{Y}$ to three points in $\mathcal{Y}$. The first point is the intersection of two lines defined by $\{h_1,h_6\}$ and $\{h_3,h_8\}$. The other two points are similarly defined. One can show that both $T_x$ and $T_d$ are invariant on $\mathcal{G}$. This is due to the fact that lines are invariant under perspective projection, therefore the image of the intersection of two lines is the intersection of the images of the two lines. However, picking an arbitrary bi-model $\{\widetilde{G}_p, \widetilde{G}_p^{-1}\}$, in general, $T_d$ and $\widetilde{G}_p^{-1} \circ \widetilde{G}$ do not commute, and $T_d$ and $\widetilde{G}_p^{-1} \circ \widetilde{G}$ do not commute. Hence, Cartesian-based approach can not guarantee precise positioning whereas both modified Cartesian-based and image-based approaches can assure precise positioning.

† It is an over-determined system equations; i.e., 3 equations and 2 unknowns. But the assumptions on $\mathcal{F}_x$ {or $\mathcal{F}_d$} guarantee that the solution $(\alpha_1, \alpha_2)$ always exists.

## 6. CONCLUSION

Three different approaches to six degree-of-freedom task encoding in vision-based control systems have been defined. These results give a partial answer to an exciting research question—how should one encode a task using sensor information to guarantee accomplishment of the original task.

Although the Cartesian-based approach does not require the invariance property, it often fails to guarantee precise positioning. Both the modified Cartesian-based and the image-based approaches are capable of guaranteeing accurate positioning provided the invariance property holds. Modified Cartesian-based approach seems to be a new idea. Its advantage and disadvantages over image-base approach is a question that needs further research. Another question that is being studied is the closed-loop performance.

## 7. REFERENCES

Chang, Wen-Chung, Joao P. Hespanha, A. S. Morse and Gregory D. Hager (1997). Task re-encoding in vision-based control systems. In: *Proc. of the 1997 Conference on Decision and Control*. IEEE. San Diego, CA, U.S.A.

Hager, Gregory D., Wen-Chung Chang and A. Stephen Morse (1995). Robot hand-eye coordination based on stereo vision. *IEEE Control Syst. Mag.* **15**(1), 30–39.

Horn (1986). *Robot Vision*. The MIT Electrical Engineering and Computer Science Series. 8 ed.. McGraw-Hill. New York.

Hutchinson, Seth A., Gregory D. Hager and Peter I. Corke (1996). A tutorial on visual servo control. *IEEE Trans. Robot. Automat.* **12**(5), 651–670.

Spong, Mark W. and M. Vidyasagar (1989). *Robot Dynamics and Control*. John Wiley & Sons, Inc.

Wilson, William J., Carol C. Williams and Grahan S. Beli (1996). Relative end-effector control using cartesian position based visual servoing. *IEEE Trans. Robot. Automat.* **12**(5), 684–696.