

# Model-Free Uncalibrated Visual Servoing Using Recursive Least Squares

Miao Hao, Peter Deuffhard, Zengqi Sun, Masakazu Fujii

Department of Computer Science and Technology, Tsinghua University, Beijing, China.

Zuse Institute Berlin (ZIB), Berlin, Germany.

Corporate Research and Development, IHI Co.,Ltd., Yokohama, Japan.

Email: haom05@mails.tsinghua.edu.cn; deuffhard@zib.de; szq-dcs@mail.tsinghua.edu.cn; masakazu\_fujii@ihi.co.jp.

**Abstract**—In this paper, a model free uncalibrated visual servoing algorithm based on recursive least squares is proposed and discussed in depth. No robot kinetics or dynamics, camera calibration or target model to be tracked would be ever needed. Based on the brief retrospection of RLS filter as well as its theoretical analysis, i.e. Wiener-Hopf condition, the core uncalibrated visual servoing algorithm based on recursive least squares is explored in detail. The experimental results of both static and moving targets applied on a Puma-560 6DOFs industrial robot simulation verifies its performances.

**Index Terms**—Visual Servoing, Jacobian Estimate, Nonlinear Control, Adaptive RLS

## I. INTRODUCTION

For numerous industrial applications of robotics and automation today, robot control with visual clues, or more technically, visual servoing, has demonstrated impressive power in large volumes of industries applications and has been regarded as one of the most promising research realms in Artificial Intelligence and Robotics [1]. Technically, the current visual servoing methods could be categorized into two main types, Position Based Visual Servoing (PBVS) and Image Based Visual Servoing (IBVS) respectively (Kragic, 2001). According to [2][3][1], most of the visual servoing systems reported in previous literature, have to utilize the robot and especially the camera model, i.e. camera calibration is needed, if satisfying results and performance required [3]. It's well known that the typical camera calibration process is in most cases elaborate, time-consuming yet not robust enough to system and environmental noises. In other word, a high accuracy camera calibration is obtained usually in constructed or at least semi constructed environment.

Therefore, the uncalibrated visual servoing without the priori knowledge of robot kinetics, dynamic as well as camera calibration was paid extensive attentions in recent years [4][5]. [6] presented uncalibrated visual servoing for static targets using fixed cameras. [7] improved the

control scheme to eye-in-hand stereo tracking of moving targets using static reference points to estimate the target motion, through the online estimation of Jacobian matrix. [8] introduced a Broyden method in non-linear least square optimization using a trust region and [9] as a review, expanded them with convergence analysis and proof. Especially, [10] proposed in detail, a dynamic visual servoing method to tracking a moving target, i.e. the velocity estimate was also done within the estimation of Jacobian matrix and compared the partitioned Broyden method with the Recursive Gauss-Newton method in depth with a 6DOF robot simulation, [9] has investigated an improved control law for moving target yet unfortunately, its convergence analysis seems incomplete which we would discussed later.

Theoretically, the best result reported in previous literature is [10] which is based on the non-linear optimization of affine invariance and adaptive algorithms, which has been utilized from [6][11] to [9][10] without much change. However, the Broyden method could be categorized as local Newton method and has to face the local minima and even stability and robustness issues. The global Newton methods have shown the preponderance versus the local Newton methods, and they are drawing more attentions in recent year [12].

This paper contributes in following three points: (1) an uncalibrated eye-in-hand visual servoing using adaptive recursive squares (VS-ARLS) versus VS-RLS is proposed, analyzed in detail and verified by a 6DOFs Puma560 simulation; (2) the evaluation criterions of the visual servoing algorithm in time-varying system are introduced; (3) the asymptotic stability analysis of the proposed algorithm in an affine contravariance perspective.

This paper is organized as follows. First of all, Section II would be a retrospection of the brief deduction of the classical uncalibrated visual servoing using recursive least squares from standard Recursive Least Square (RLS), thereby introducing the performance evaluation criterions of time-varying system. Based on that, the uncalibrated visual servoing using adaptive recursive least square (VS-ARLS) is proposed in Section III and deduced from adaptive recursive least squares (ARLS). Moreover, the asymptotic stability of the proposed algorithm is proved in an affine contravariance perspective in Section IV.

This paper is based on "Uncalibrated Visual Servoing Using Recursive Least Squares," by M. Hao, Z.Q. Sun, M. Fujii, W. Song, which appeared in the Proceedings of the 2007 IEEE International Conference on Systems, Man and Cybernetics (SMC), Montreal, Canada, October 2007. © 2007 IEEE.

This work is supported by the international collaboration project with IHI, Japan, and also in part by the 973 High Tech. Programs of China (2002CB312205).

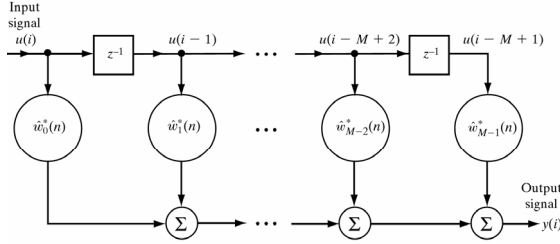


Figure 1. The structure of an RLS filter. This figure is referred to [13].

Besides, experimental results would be discussed in Section V and performance analysis in Section VI. Finally, Section VII would be the conclusive section summarizing and indicating the research topics in the future.

## II. CLASSICAL UNCALIBRATED VISUAL SERVOING USING RECURSIVE LEAST SQUARES

In this section, the classical uncalibrated visual servoing using recursive least squares is introduced and discussed briefly, for the incoming comparison of the new algorithm using adaptive recursive least squares.

### A. Recursive Least Squares

It's constructive to have a retrospection of Recursive Least Squares (RLS) from which the classical VS-RLS would be deduced. A structural view of RLS filter is shown in Fig. 1.

Since the observable data is variable, the cost function to be minimized could be expressed as  $\mathbf{E}(n)$  where  $n$  is the variable length of the observation data. And the cost function is often expressed in a weighting factor way, i.e.

$$\mathbf{E}(n) = \sum_{i=1}^n \beta(n, i) |e(i)|^2 \quad (1)$$

where  $\beta(i)$  is the filter coefficient and  $e(i)$  the error between desired response  $d(i)$  and current output  $y(i)$ , i.e.

$$\begin{aligned} e(i) &= d(i) - y(i) \\ &= d(i) - \mathbf{w}^H(n)\mathbf{u}(n) \end{aligned} \quad (2)$$

and

$$\begin{aligned} \mathbf{u}(i) &= [u(i), u(i-1), \dots, u(i-M+1)]^T \\ \mathbf{w}(n) &= [w_0(n), w_1(n), \dots, w_{M-1}(n)]^T \end{aligned} \quad (3)$$

where  $\mathbf{u}(i)$  is the input signal vector and  $\mathbf{w}(n)$  the weighting vector.

Let

$$\beta(n, i) = \lambda^{n-i}, i = 1, 2, \dots, n \quad (4)$$

where  $\lambda$  is noted as forgetting factor to forget the past information. The optimal solution of (1) is then equal to the following Wiener-Hopf equation

$$\Phi(n)\tilde{\mathbf{w}}(n) = \mathbf{z}(n) \quad (5)$$

where

$$\begin{aligned} \Phi(n) &= \sum_{i=1}^n \lambda^{n-i} \mathbf{u}(i)\mathbf{u}^H(i) \\ \mathbf{z}(n) &= \sum_{i=1}^n \lambda^{n-i} \mathbf{u}(i)d^*(i) \end{aligned} \quad (6)$$

which could be re-written in recursive form and expanded using matrix inversion lemma, then we immediately have the exponentially weighted recursive least squares algorithm [13]

### Initialization:

$\tilde{\mathbf{w}}(0) = \mathbf{0}$ ;  $\mathbf{P} = \delta^{-1}\mathbf{I}$  is a small positive constant.

### Iteration body: for $n=1,2,\dots,N$

$$\begin{aligned} \mathbf{k}(n) &= \frac{\lambda^{-1}\mathbf{P}(n-1)\mathbf{u}(n)}{1 + \lambda^{-1}\mathbf{u}^H(n)\mathbf{P}(n-1)\mathbf{u}(n)} \\ \xi(n) &= d(n) - \mathbf{w}^H(n-1)\mathbf{u}(n) \\ \mathbf{w}(n) &= \mathbf{w}(n-1) + \mathbf{k}(n)\xi(n) \\ \mathbf{P}(n) &= \lambda^{-1}\mathbf{P}(n-1) - \lambda^{-1}\mathbf{k}(n)\mathbf{u}^H(n)\mathbf{P}(n-1) \end{aligned} \quad (7)$$

### B. Uncalibrated Eye-in-hand Visual Servoing Using Recursive Least Squares

The task description of the uncalibrated eye-in-hand visual servoing could be described as follows [12]: given a robot  $\mathbb{R}$  and an eye-in-hand camera  $\mathbb{C}$  without calibration, the objective is to move the robot from the eye-in-hand image feature  $\mathbf{y}(\theta)$  under current joint value  $\theta$ , to the desired image feature  $\mathbf{y}^*(\theta)$ . No robot kinetics or dynamics model needed. Nor does the camera model.

The error function is defined as

$$\mathbf{f}(\theta, \mathbf{t}) = \mathbf{y}(\theta) - \mathbf{y}^*(\mathbf{t}) \quad (8)$$

The squared error function is defined as

$$\mathbf{F}(\theta, \mathbf{t}) = \frac{1}{2}\mathbf{f}^T(\theta, \mathbf{t})\mathbf{f}(\theta, \mathbf{t}) \quad (9)$$

And the squared error function could be expressed in first order Taylor form as

$$\mathbf{F}(\theta + \Delta\theta, \mathbf{t} + \Delta\mathbf{t}) = \mathbf{F}(\theta, \mathbf{t}) + \frac{\partial \mathbf{F}}{\partial \theta} \Delta\theta + \frac{\partial \mathbf{F}}{\partial \mathbf{t}} \Delta\mathbf{t} + \mathbf{O}(\Delta\theta, \Delta\mathbf{t}) \quad (10)$$

considering that the sampling period is assumed to be a constant, the minimum of  $\mathbf{F}(\theta, \mathbf{t})$  could be calculated as

$$\frac{\partial \mathbf{F}(\theta + \Delta\theta, \mathbf{t} + \Delta\mathbf{t})}{\partial \theta} = 0 \quad (11)$$

Omit the high order infinite population  $\mathbf{O}(\Delta\theta, \Delta\mathbf{t})$ , approximately we have

$$\frac{\partial \mathbf{F}}{\partial \theta} + \frac{\partial^2 \mathbf{F}}{\partial \theta^2} \Delta\theta + \frac{\partial^2 \mathbf{F}}{\partial \theta \partial \mathbf{t}} \Delta\mathbf{t} = 0 \quad (12)$$

then

$$\Delta\theta = - \left( \frac{\partial^2 \mathbf{F}}{\partial \theta^2} \right)^{-1} \left( \frac{\partial \mathbf{F}}{\partial \theta} + \frac{\partial^2 \mathbf{F}}{\partial \theta \partial \mathbf{t}} \Delta\mathbf{t} \right) \quad (13)$$

Substituting (13) with (9) and (10) would result in

$$\Delta\theta = -(\mathbf{J}^T \mathbf{J} + \mathbf{S})^{-1} \mathbf{J}^T \left( \mathbf{f} + \frac{\partial \mathbf{f}}{\partial t} \Delta t \right) \quad (14)$$

where

$$\mathbf{J} = \frac{\partial \mathbf{f}}{\partial \theta}, \mathbf{S} = \frac{\partial \mathbf{J}^T}{\partial \theta} \mathbf{f}, \frac{\partial \mathbf{F}}{\partial \theta} = \mathbf{J}^T \mathbf{f}, \quad (15)$$

$$\frac{\partial^2 \mathbf{F}}{\partial \theta \partial t} = \mathbf{J}^T \frac{\partial \mathbf{f}}{\partial t}, \frac{\partial^2 \mathbf{F}}{\partial \theta^2} = \mathbf{J}^T \mathbf{J} + \mathbf{S},$$

Note the definition  $\mathbf{J} = \partial \mathbf{f} / \partial \theta$  is known as composite Jacobian matrix. Moreover, since  $\mathbf{S}$  is system dependent and actually difficult to estimate, yet it could be regarded as an infinite population, because when the robot position is near the desired, the error function  $\mathbf{f}(\theta, \mathbf{t})$  could be regarded as zero. Therefore, we could re-write (15) as

$$\Delta\theta = -\left( \tilde{\mathbf{J}}_k^T \tilde{\mathbf{J}}_k \right)^{-1} \tilde{\mathbf{J}}_k^T \left( \mathbf{f} + \frac{\partial \mathbf{f}}{\partial t} \Delta t \right) \quad (16)$$

where  $\tilde{\mathbf{J}}$  stands for the estimation of the composite Jacobian matrix at the  $K$ th iteration. And (16) is the joint value update formula.

Besides, the affine model of error function is defined as

$$\mathbf{m}_k(\theta, t) = \mathbf{f}(\theta_k, t_k) + \tilde{\mathbf{J}}_k(\theta - \theta_k) + \frac{\partial \mathbf{f}_k}{\partial t}(t - t_k) \quad (17)$$

which could be regarded as also the first order expansion of  $\mathbf{m}_k(\theta, t)$  at  $(\theta_k, t_k)$  and the corresponding target function is

$$\min \mathbf{E}(k) = \sum_{i=0}^{k-1} \lambda^{k-i-1} \|\Delta \mathbf{m}_{ki}\|^2 \quad (18)$$

$$\Delta \mathbf{m}_{ki} = \mathbf{m}_k(\theta_i, t_i) - \mathbf{m}_i(\theta_i, t_i)$$

$$= [\mathbf{f}_k - \mathbf{f}_i - \frac{\partial \mathbf{f}_k}{\partial t}(t_k - t_i)] - \mathbf{J}_k \mathbf{h}_{ki}$$

$$= [\mathbf{f}_k - \mathbf{f}_i] - \mathbf{J}_k \mathbf{h}_{ki}$$

where we denote

$$\mathbf{h}_{ki} = (\theta_k - \theta_i) \quad (19)$$

$$h_k = t_{k+1} - t_k$$

Now it's time to look backward at the RLS algorithm. Compare (18) to (7) with (1) and (4) in mind, if we make the following substitutions

$$d \rightarrow \Delta \mathbf{f} = \mathbf{f}_k - \mathbf{f}_{k-1}$$

$$\mathbf{w}^H \rightarrow \mathbf{J}_k = \begin{bmatrix} \tilde{\mathbf{J}}_k & (\tilde{\mathbf{f}}_t)_{k-1} \end{bmatrix} \quad (20)$$

$$\mathbf{u} \rightarrow \mathbf{h} = \begin{bmatrix} \theta_k - \theta_{k-1} \\ t_k - t_{k-1} \end{bmatrix}$$

where scalar expanded to a vector, and column vector to rectangle matrix, then the complete description of the VS-RLS algorithm could be expressed as [10][12]

---

#### Initialization:

Given  $f: \mathbb{R}^n \rightarrow \mathbb{R}^m; \theta_0, \theta_1 \in \mathbb{R}^{m \times n};$   
 $(\tilde{f}_t)_0 \in \mathbb{R}^{m \times 1}; P_0 \in \mathbb{R}^{n+1 \times n+1}; \lambda \in (0, 1];$

---

#### Iteration body:

$$\Delta \mathbf{f} = \mathbf{f}_k - \mathbf{f}_{k-1}; \mathbf{h}_\theta = \theta_k - \theta_{k-1}; h_t = t_k - t_{k-1}$$

$$\mathbf{h} = \begin{bmatrix} \theta_k - \theta_{k-1} \\ t_k - t_{k-1} \end{bmatrix}$$

$$\mathbf{J}_k = \begin{bmatrix} \tilde{\mathbf{J}}_k & (\tilde{\mathbf{f}}_t)_{k-1} \end{bmatrix}$$

$$\Delta \mathbf{J}_k = (\Delta \mathbf{f} - \mathbf{J}_{k-1} \mathbf{h}) (\lambda + \mathbf{h}^T \mathbf{P}_{k-1} \mathbf{h})^{-1} \mathbf{h}^T \mathbf{P}_{k-1}$$

$$\mathbf{J}_k = \mathbf{J}_{k-1} + \Delta \mathbf{J}_k$$

$$\mathbf{P}_k = \frac{1}{\lambda} \left[ \mathbf{P}_{k-1} - \frac{\mathbf{P}_{k-1} \mathbf{h} \mathbf{h}^T \mathbf{P}_{k-1}}{\lambda + \mathbf{h}^T \mathbf{P}_{k-1} \mathbf{h}} \right]$$

$$\theta_{k+1} = \theta_k - \left( \tilde{\mathbf{J}}_k^T \tilde{\mathbf{J}}_k \right)^{-1} \left( \tilde{\mathbf{J}}_k^T \mathbf{f}_k + \tilde{\mathbf{J}}_k^T (\tilde{\mathbf{f}}_t)_k \mathbf{h} \right) \quad (21)$$


---

### III. PERFORMANCE EVALUATION OF TIME-VARYING SYSTEM TRACKING

For stationary environment where both the statistical variables and the error performance surface are fixed, the transient or natural gradient based method in most cases would result in an optimum or at least near optimum performance. Yet when it comes to a time-vary system where the statistical description of the system becomes non-stationary or dynamic, the error performance surface is no longer fixed and therefore a new evaluation of tracking performance is added.

As [13] stated, tracking is a steady-state phenomenon, yet the statistical variations of the environment has to be slow enough for tracking to be feasible. Then the concept of degree of nonstationarity is introduced as [14]

$$\alpha = \left( \frac{\mathbf{E} \left[ |\mathbf{w}^H(n) \mathbf{u}(n)|^2 \right]}{\mathbf{E} \left[ |v(n)|^2 \right]} \right)^{1/2} \quad (22)$$

where  $\mathbf{w}(n)$  and  $v(n)$  are process noise vector and measurement noise respectively.

Technically, the criterions for tracking evaluation could be categorized as follows: mean-square deviation and displacement respectively.

#### A. Mean Square Deviation

One of the most commonly used merit for tracking evaluation is Mean Square Deviation (MSD), defined as

$$\mathcal{D}(n) = \mathbf{E} \left[ |\hat{\mathbf{w}}(n) - \mathbf{w}_0(n)|^2 \right] \quad (23)$$

where

$$\mathcal{D}(n) = \mathcal{D}_1(n) + \mathcal{D}_2(n)$$

$$\mathcal{D}_1(n) = \mathbf{E} \left[ |\tilde{w}(n) - \mathbf{E}[\tilde{\mathbf{w}}(n)]|^2 \right] \quad (24)$$

$$\mathcal{D}_2(n) = \mathbf{E} \left[ |\mathbf{E}[\tilde{\mathbf{w}}(n)] - \mathbf{w}_0(n)|^2 \right]$$

and  $\mathcal{D}_1(n)$  is denoted as estimation variance due to the weight vector noise and  $\mathcal{D}_2(n)$  is called lag variance which is the testimony to the nonstationary nature of the environment.

**B. Replacement**

Another widely used criterion is misplacement define as

$$\mathfrak{M} = \frac{J_{ex}(n)}{\sigma_v^2} \quad (25)$$

where  $J_{ex}(n)$  is the excess mean squared error of adaptive filter measured with respect to the variance  $\sigma_v^2$ . Again, the displacement could also be divided into two components as

$$\mathfrak{M}(n) = \mathfrak{M}_1(n) + \mathfrak{M}_2(n) \quad (26)$$

where  $\mathfrak{M}_1(n)$  is called noise misadjustment and the second lag misadjustment, in that the estimation noise and lag noise are decoupled in power.

**IV. CLASSICAL UNCALIBRATED VISUAL SERVOING USING ADAPTIVE RECURSIVE LEAST SQUARES**

Taking the discussions and performance evaluation criterions discussed in Section III in mind, one aspect of promoting the performance of system tracking in a RLS form, which is the very case stated in this paper, is to focus on the adaptive tuning of the learning rate  $\lambda$ , i.e. to select the optimal learning rate in every step.

Again, the cost function could be re-defined as

$$J'(n) = \frac{1}{2} E[|\xi(n)|^2] \quad (27)$$

where  $\xi(n)$  is the priori knowledge error, i.e.

$$\xi(n) = d(n) - \mathbf{w}^H(n-1)\mathbf{u}(n) \quad (28)$$

The partial  $J'(n)$ , partial  $\lambda$  is

$$\begin{aligned} \nabla \lambda(n) &= \frac{\partial J'(n)}{\partial \lambda} \\ &= \frac{1}{2} E \left[ \frac{\partial \xi(n)}{\partial \lambda} \xi^*(n) + \frac{\partial \xi^*(n)}{\partial \lambda} \xi(n) \right] \\ &= -\frac{1}{2} E \left[ \Psi^H(n-1)\mathbf{u}(n)\xi^*(n) + \mathbf{u}^H(n)\Psi(n-1)\xi(n) \right] \end{aligned} \quad (29)$$

where

$$\Psi(n) = \frac{\partial \mathbf{w}(n)}{\partial \lambda} \quad (30)$$

Note that in (7) we have

$$\begin{aligned} \mathbf{k}(n) &= \mathbf{P}(n)\mathbf{u}(n) \\ \mathbf{w}(n) &= \mathbf{w}(n-1) + \mathbf{P}(n)\mathbf{u}(n)\xi^*(n) \end{aligned} \quad (31)$$

and if we denote

$$\mathbf{S}(n) = \frac{\partial \mathbf{P}(n)}{\partial \lambda} \quad (32)$$

then it's easy to have

$$\begin{aligned} \Psi(n) &= [\mathbf{I} - \mathbf{k}(n)\mathbf{u}^H(n)]\Psi(n-1) + \mathbf{S}(n)\mathbf{u}(n)\xi^*(n) \\ \mathbf{P}(n) &= \lambda^{-1}\mathbf{P}(n-1) - \frac{\lambda^{-2}\mathbf{P}(n-1)\mathbf{u}(n)\mathbf{u}^H(n)\mathbf{P}(n-1)}{1+\lambda^{-1}\mathbf{u}^H(n)\mathbf{P}(n-1)\mathbf{u}(n)} \end{aligned} \quad (33)$$

Differentiating the second equation in (29) yields

$$\begin{aligned} \mathbf{S}(n) &= \lambda^{-1}[\mathbf{I} - \mathbf{k}(n)\mathbf{u}^H(n)]\mathbf{S}(n-1)[\mathbf{I} - \mathbf{u}(n)\mathbf{k}^H(n)] \\ &\quad + \lambda^{-1}\mathbf{k}(n)\mathbf{k}^H(n) - \lambda^{-1}\mathbf{P}(n) \end{aligned} \quad (34)$$

Now we are ready to deduce the adaptive learning rate  $\lambda$ . If we substitute the scalar gradient  $\nabla \lambda(n)$  with instantaneous gradient estimation  $-\text{Re}[\Psi^H(n-1)\mathbf{u}(n)\xi^*(n)]$ , the adaptively tuned learning rate  $\lambda$  is calculated as

$$\begin{aligned} \lambda(n) &= \lambda(n-1) - \alpha \widehat{\nabla} \lambda(n) \\ &= \lambda(n-1) + \alpha \text{Re}[\Psi^H(n-1)\mathbf{u}(n)\xi^*(n)] \end{aligned} \quad (35)$$

where  $\alpha$  is a small positive parameter. Now we've successfully expand the standard RLS algorithm into adaptive recursive least squares (ARLS) where actually fixed  $\lambda$  is expanded to time-varying  $\lambda(n)$ . The ARLS could be described as follows

**Initialization:**

$\tilde{\mathbf{w}}(0) = \mathbf{0}$ ;  $\mathbf{P} = \delta^{-1}\mathbf{I}$  is a small positive constant.

**Iteration body: for**  $n=1,2,\dots,N$

$$\begin{aligned} \mathbf{k}(n) &= \frac{\lambda^{-1}\mathbf{P}(n-1)\mathbf{u}(n)}{1+\lambda^{-1}\mathbf{u}^H(n)\mathbf{P}(n-1)\mathbf{u}(n)} \\ \xi(n) &= d(n) - \mathbf{w}^H(n-1)\mathbf{u}(n) \\ \mathbf{w}(n) &= \mathbf{w}(n-1) + \mathbf{k}(n)\xi(n) \\ \mathbf{P}(n) &= \lambda^{-1}\mathbf{P}(n-1) - \lambda^{-1}\mathbf{k}(n)\mathbf{u}^H(n)\mathbf{P}(n-1) \\ \lambda(n) &= [\lambda(n-1) + \alpha \text{Re}[\Psi^H(n-1)\mathbf{u}(n)\xi^*(n)]]_{\lambda_-}^{\lambda_+} \\ \mathbf{S}(n) &= \lambda^{-1}(n)[\mathbf{I} - \mathbf{k}(n)\mathbf{u}^H(n)]\mathbf{S}(n-1)[\mathbf{I} - \mathbf{u}(n)\mathbf{k}^H(n)] \\ &\quad + \lambda^{-1}(n)\mathbf{k}(n)\mathbf{k}^H(n) - \lambda^{-1}\mathbf{P}(n) \\ \Psi(n) &= [\mathbf{I} - \mathbf{k}(n)\mathbf{u}^H(n)]\Psi(n-1) + \mathbf{S}(n)\mathbf{u}(n)\xi^*(n) \end{aligned} \quad (36)$$

where  $\lambda_+$  and  $\lambda_-$  stand for upper and lower bound truncation respectively and its equation typical Newton Leibniz formula. Empirically,  $\lambda_+$  is preferred to near 1 yet  $\lambda_-$  plays a critical role in the tracking performance.

Similarly, we would have immediately the uncalibrated eye-in-hand visual servoing using adaptive recursive least squares (VS-RLS) as

**Initialization:**

$f: \mathbb{R}^n \rightarrow \mathbb{R}^m$ ;  $\theta_0, \theta_1 \in \mathbb{R}^{m \times n}$ ;  
 $(\tilde{f}_t)_0 \in \mathbb{R}^{m \times 1}$ ;  $P_0 \in \mathbb{R}^{(n+1) \times (n+1)}$ ;  $\lambda \in (0, 1]$ ;

**Iteration body:**

$$\begin{aligned} \Delta \mathbf{f}_k &= \mathbf{f}_k - \mathbf{f}_{k-1}; \mathbf{h}_\theta = \theta_k - \theta_{k-1}; h_t = t_k - t_{k-1} \\ \mathbf{h} &= \begin{bmatrix} \theta_k - \theta_{k-1} \\ t_k - t_{k-1} \end{bmatrix} \\ \mathbf{J}_k &= [\tilde{\mathbf{J}}_k = (\tilde{\mathbf{f}}_t)_{k-1}] \\ \Delta \mathbf{J}_k &= (\Delta \mathbf{f}_k - \mathbf{J}_{k-1}\mathbf{h})(\lambda_n + \mathbf{h}^T \mathbf{P}_{k-1} \mathbf{h})^{-1} \mathbf{h}^T \mathbf{P}_{k-1} \\ \mathbf{J}_k &= \mathbf{J}_{k-1} + \Delta \mathbf{J}_k \\ \mathbf{P}_k &= \frac{1}{\lambda_n} \left[ \mathbf{P}_{k-1} - \frac{\mathbf{P}_{k-1} \mathbf{h} \mathbf{h}^T \mathbf{P}_{k-1}}{\lambda + \mathbf{h}^T \mathbf{P}_{k-1} \mathbf{h}} \right] \\ \theta_{k+1} &= \theta_k - \left( \tilde{\mathbf{J}}_k^T \tilde{\mathbf{J}}_k \right)^{-1} \left( \tilde{\mathbf{J}}_k^T \mathbf{f}_k + \tilde{\mathbf{J}}_k^T (\tilde{\mathbf{f}}_t)_k \mathbf{h} \right) \\ \lambda_n &= [\lambda_{n-1} + \alpha (\Delta \mathbf{f}_k - \mathbf{J}_{k-1}\mathbf{h})' \Psi_n \mathbf{h}]_{\lambda_-}^{\lambda_+} \\ \mathbf{S}_n &= \lambda_n^{-1} [\mathbf{I} - \mathbf{k}_n \mathbf{h}' ] \mathbf{S}_{n-1} [\mathbf{I} - \mathbf{h} \mathbf{k}_n^H] \\ &\quad + \lambda_n^{-1} \mathbf{k}_n \mathbf{k}_n^H - \lambda_n^{-1} \mathbf{P}_n \\ \Psi_n &= [\mathbf{I} - \mathbf{k}_n \mathbf{u}_n^H] \Psi_{n-1} + (\Delta \mathbf{f}_k - \mathbf{J}_{k-1}\mathbf{h})(\mathbf{S}_n \mathbf{h})' \end{aligned} \quad (37)$$

**V. EXPERIMENTAL SIMULATIONS**

In this section, a 6DOF Puma560 robot simulation using Robotics Toolbox in MatLab [15] is explored and

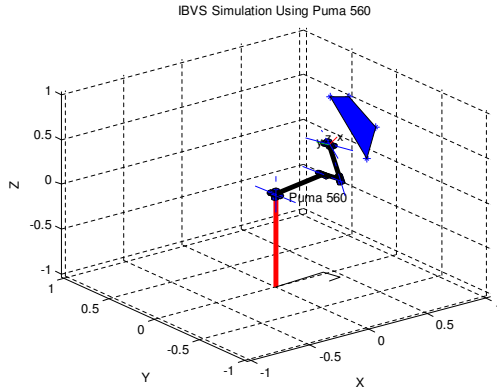


Figure 2. The simulation of 6DOF Puma560 using Robotics Toolbox for MatLab (Corke, 1996) as well as the target in world coordination system.

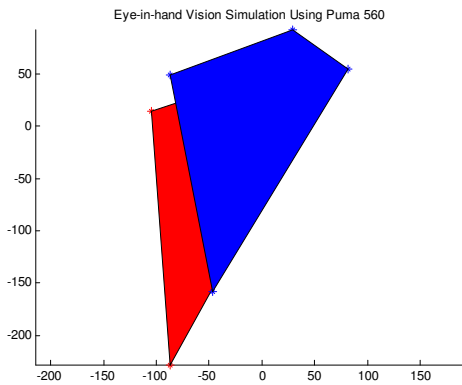


Figure 3. The eye-in-hand vision of Puma560 where the red region indicates the desired image feature and the blue the current.

discussed. The eye-in-hand camera is assumed to be coincident with the final joint, i.e. the end grasper pose and position is those of the camera. The robot simulation toolbox is shown in Fig. 2, including the initial robot pose and position as well as the target.

The sampling period is 50ms and the system runs at 20Hz. The target is selected as an irregular quadrangle whose homogeneous world coordination vector is

$$P_W(A, B, C, D) = \begin{bmatrix} 0.700 & 0.700 & 0.700 & 0.700 \\ -0.100 & 0.300 & 0.100 & -0.200 \\ 0.200 & 0.700 & 0.800 & 0.600 \\ 1.000 & 1.000 & 1.000 & 1.000 \end{bmatrix} \quad (38)$$

Note that the selection of this quadrangle is to avoid the potential singularity of Jacobian matrix. Experimental results in [9][10] are of square targets yet the desired feature vector in eye-in-hand image plane is not, for the same reason.

The focus length is  $f = 0.004m$  and the pixel projection ratio is  $P = 30,000\text{pixel}/m$ . The static and moving target tracking are done respectively. The additive uniform noise to the coordination of the image plane is 0.2 pixel.

Let the Z axis and Y axis in the end grasper coordination system be coincidental with the X and Y axis in the eye-in-hand image plane, the coordination translation

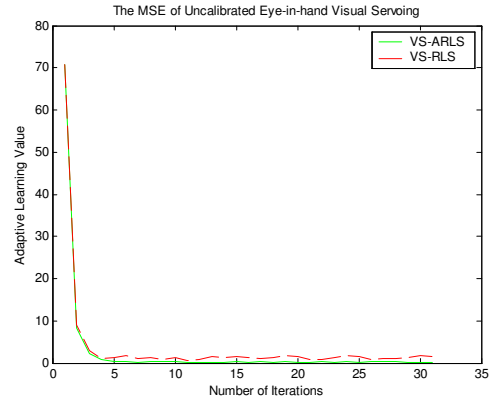


Figure 4. The MSE curves of static target tracking using VS-RLS and VS-ARLS. The solid line is VS-ARLS and the dotted VS-RLS. The average of MSE using VS-ARLS is 1.863.

from  $P(x, y, z)$ , the end grasper coordination system to the image plane system  $(I_x, I_y)$  is

$$\begin{bmatrix} I_x \\ I_y \end{bmatrix} = \frac{f \cdot P}{x} \begin{bmatrix} y \\ z \end{bmatrix} \quad (39)$$

#### A. Initial Jacobian Matrix Estimation

The initial Jacobian matrix estimation is believed to impact dramatically the performance of the algorithm, i.e. if improperly chosen, the servoing process does not even work and the robot seems out of control.

In [16], the tiny displacements of each joint and the corresponding displacements of the feature vector consist of  $n$  pairs

$$[(d\theta_0, d\mathbf{f}_0), (d\theta_1, d\mathbf{f}_1), \dots, (d\theta_n, d\mathbf{f}_n)] \quad (40)$$

and then we have

$$\begin{aligned} \Theta_D &= [d\theta_0, d\theta_1, \dots, d\theta_n] \\ \mathbf{F}_D &= [d\mathbf{f}_0, d\mathbf{f}_1, \dots, d\mathbf{f}_n] \end{aligned} \quad (41)$$

Now the initial Jacobian matrix estimation is calculated as

$$\tilde{\mathbf{J}}_0 = \mathbf{F}_D (\Theta_D)^{-1} \quad (42)$$

#### B. Static Target Tracking

The curves of MSE and joint value of static target tracking are shown in Fig. 4 and Fig. 5 respectively. The convergence speeds of VS-ARLS and VS-RLS are nearly same, yet the average MSE of VS-ARLS is better than that of VS-RLS. The average MSE of VS-ARLS in static target tracking is 1.863pixels.

#### C. Dynamic Target Tracking

The ellipse motion curve of the target is

$$\begin{cases} y = -0.1 + 0.1 \sin(1 * h_t * t_k) \\ z = 0.1 + 0.2 \cos(1 * h_t * t_k) \end{cases} \quad (43)$$

Then the curves of MSE and joint value of moving target tracking are as shown in Figure-6 and Figure-7 respectively. We could see that the controller of VS-ARLS performs fairly well than VS-RLS and could be able to track the moving target. The mean tracking MSE is 2.863.

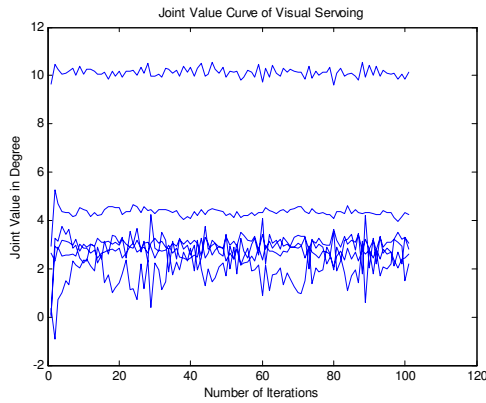


Figure 5. The joint value curves of static target tracking of VS-ARLS.

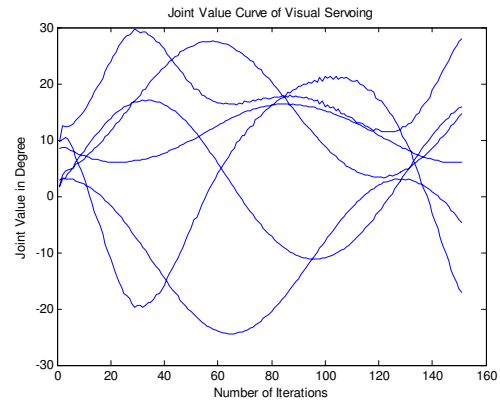


Figure 7. The joint value curves of moving target tracking of VS-ARLS.

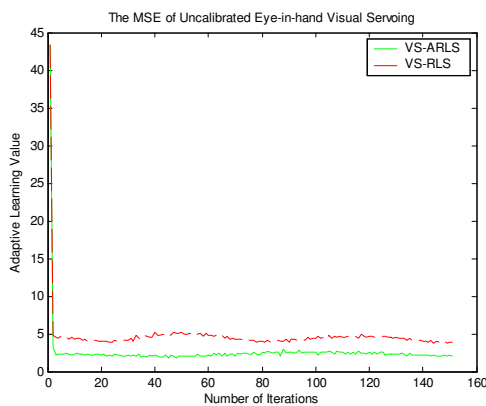


Figure 6. The MSE curves of moving target tracking using VS-RLS and VS-ARLS. The solid line is VS-ARLS and the dotted VS-RLS. The average of MSE using VS-ARLS is 2.863.

## VI. PERFORMANCE ANALYSIS AND DISCUSSION

In this section, the performance of the proposed algorithm is analyzed critically, in a pros and cons manner. We could summary this problem in multiple perspectives. Theoretically, this algorithm could be categorized into residual based affine invariance/contravariance Newton algorithms in terms of Numerical Analysis, or an online estimation of a highly nonlinear, high dimensional, ill-conditioned and even time-varying (if tracking moving target) kernel matrix, the Jacobian matrix denoted as in terms of System Identification or, a Finite Impulse Response filter to attack an augmented tap vector, which is exactly the Jacobian matrix applied to "filter" the robot joint value vector as input and output the image feature vector or finally, a Visual Servoing issue to drive the robot arm to a desired pose and position in terms of Robotics, the most initiative point of view.

### A. The pros of this algorithm

For a model free uncalibrated visual servoing algorithm, its performance demonstrated is satisfying and its pros are easy to have: (1) do not need robot kinetics or dynamics; (2) do not need the all consuming yet delicate process camera calibration; (3) is capable to track any target where current and desired feature vector could be extracted; (4) do not need any pre or pose process such as system modeling or environmental interaction.

### B. The cons of this algorithm

Yet we would like to concentrate more on its cons, which might be brought along as all the pro items, turning peach into lemon.

(1) Local minima is inevitable for a inherently local Newton based algorithm. As well known to all, local Newton algorithm would suffer trapping in a small gradient zone anyway. Technically, this algorithm tries to climb the complicated shaped and full-of-pitfall hill anyway. Whenever the transient gradient estimation is not that accurate, it snares you.

(2) Moreover, as same to most local Newton algorithms, it's initial estimation sensitive, i.e., a "good enough" initial estimation of the kernel Jacobian matrix is crucial for a successful visual servoing process. Attacking (1) and (2) simultaneously would result in a global expansion of the current algorithm, one of the most promising improvements. Note that the construction of global Newton based visual servoing algorithm, which is capable to track the target more flexibly, is strongly recommended in the affine invariance framework introduced for the very reason that the theoretical analysis of affine invariance algorithms would directly lead to construction of algorithms.

(3) Most of its pros could be regarded as autonomous system identification or filtering, yet frankly speaking it's quite a challenge for a classical RLS filter to estimate such a highly nonlinear and ill-conditioned kernel. In such a situation, it is more vulnerable to system modeling errors and noises. Since it has to utilize every bit of information obtained from each step contaminated by system modeling and noises, over-fitting is some extent possible where the online adjustment of the kernel would reluctantly influenced by noises. To overcome that, tradeoff between having more information to learn and resisting the noise and error contamination has to be made, resulting in empirical adjustment of the forgetting factor. Therefore, a more explicit function linking the error-and-noise and the forgetting factor would be constructive to achieve better performance.

(4) When it comes to a more general issue of online system identification, the singularity of Jacobian matrix is another challenge that needs to shoot. Since most

industrial robot applied consist of rotational joints, the Jacobian matrix runs into singular or at least highly ill-conditioned in extreme cases such as the adjacent two rods aligns, let alone all kinds of 6DOFs robots. That's partial reason of its ill-condition. To solve that inherent problem is substantially challenging, yet a natural choice is to introduce an artificial potential field like method to circumvent this issue, that is, when the robot runs into a highly ill-conditioned pose or position, the weighted potential field would hinder it from moving forward to further singular zone.

#### ACKNOWLEDGMENT

The corresponding author of this paper sincerely thanks to Prof./Dr. H.Lipkin as well as Dr. J.A.Piepmeyer for fruitful and constructive discussion with the topic. Besides, he also acknowledges the anonymous reviewer(s) of the Journal of Computers for penetrating comments on this paper.

#### REFERENCES

- [1] Hutchinson S., Hager G D, Corke P I, A Tutorial on Visual Servo Control, Robotics and Automation, IEEE Transaction on, Vol. 12, 651-670, Oct, 1996
- [2] Kragic D., Visual Servoing for Manipulation: Robustness and Integration Issues, Ph.D. Dissertation, Royal Institute of Technology, Sweden, 2001
- [3] Deng L.F., Comparison of Image-Based and Position-Based Robot Visual Servoing Methods and Improvements, Ph.D. Dissertation, Waterloo University, Canada, 2003
- [4] Jang W., Bien Z., Feature-based Visual Servoing of an Eye-in-hand Robot with Improved Tracking Performance, IEEE International Conference on Robotics and Automation, Vol. 3, 2254-2260, Sacramento, CA, USA, 1991
- [5] Malis E., Chaumette F., Theoretical Improvements in the Stability Analysis of a New Class of Model-free Visual Servoing Methods. Robotics and Automation, IEEE Transactions on, Vol. 18(2), 176-186, 2002
- [6] Hosoda K., Igarashi K., Asada M., Adaptive hybrid control for visual servoing and force servoing in an unknown environment, IEEE Robot. Automat. Mag., Vol. 5(4), 39-43, 1998
- [7] Tanaka T., Asada M., Hosoda K., Visual tracking of unknown moving object by adaptive binocular visual servoing, Proc. IEEE Int. Conf. Multisensor Fusion and Integration for Intelligent Systems, Vol. 3, 2076-2081, San Francisco, USA, 1999
- [8] Jagerand M., Fuentes O., Nelson R., Experimental Evaluation of Uncalibrated Visual Servoing for Precision Manipulation. Proceedings of IEEE International Conference on Robotics and Automation, Vol. 4, 2874-2880, Albuquerque, NM, USA, 1997
- [9] Piepmeyer J. A., McMurray G. V., Lipkin H., Uncalibrated Dynamic Visual Servoing, Robotics, IEEE Transactions on, Vol. 20(1), 143-147, Feb, 2004
- [10] Piepmeyer J.A., Lipkin H., Uncalibrated Eye-in-hand Visual Servoing, The International Journal of Robotics Research, Vol. 22, 805-819, 2003
- [11] Hosoda K., Asada M., Versatile visual servoing without knowledge of true Jacobian, Proc. IEEE/RSJ/GI Int. Conf. Intelligent Robots and Systems, Vol. 1, 186-193, Munich, Germany, Sept. 1994
- [12] Hao M., Sun Z.Q., Fujii M., Song W., Uncalibrated Eye-in-hand Visual Servoing Using Recursive Least Squares, Proc. IEEE Int. Conf. System, Man and Cybernetics, 64-69, Montreal, Canada, 2007
- [13] Haykin S., Adaptive Filter Theory 3rd, Englewood Cliffs, Prentice-Hall, New York, USA, 2004
- [14] Macchi O., Adaptive Processing: The LMS Approach with Applications in Transmission, Wiley, New York, USA, 1995
- [15] Corke P.I., A Robotics Toolbox for MatLab, IEEE Robotics and Automation Magazine., Vol. 3(1), 24-32, Mar, 1996
- [16] Zhou L., Guo Z.M., Application of Exploratory Motion in Visual Servoing Method without Calibration, Journal Harbin Univ. Sci.&Tech., Vol. 7(1), 11-13, Feb, 2002 (in Chinese)
- [17] Deuffhard P., Newton Methods for Nonlinear Problems: Affine Invariance and Adaptive Algorithms, Springer-Verlag, Berlin, Germany, 2004
- [18] Cretual, A., Chaumette, F., Bouthemy, P., Complex Object Tracking by Visual Servoing Based on 2d Image Motion, 14th International Conference on Pattern Recognition, Vol. 2, 1251-1254, Brisbane, Qld., Australia, 1991
- [19] Flandin G., Chaumette F., Marchand E., Eye-in-hand/eye-to-hand cooperation for visual servoing, IEEE International Conference on Robotics and Automation, Vol. 3, 2741-2746, San Francisco, USA, 2000
- [20] Hao M., Sun Z.Q., Uncalibrated Eye-in-hand Visual Servoing Using Adaptive Recursive Least Squares, 17th World Congress of International Federation of Automatic Control, 1021-1026, Chongqing, China, 2008.
- [21] Zhao Q.J., Lian G.Y., Sun Z.Q., Survey of Robot Visual Servoing, Control and Decision, Nov, 2001 (in Chinese)

#### APPENDIX

Historically, the first classical convergence theorems for Newton series methods are Newton-Kantorovich theorem and Newton-Mysovskikh theorem respectively. Newton-Kantorovich theorem introduces Kantorovich quantity  $h_0 = \|\Delta x_0\|_X \beta_0 \gamma < \frac{1}{2}$  and a convergence ball round  $x_0$  with radius  $\rho_0 \sim 1/\beta_0 \gamma$ . Similarly, Newton-Mysovskikh theorem introduces Mysovskikh quantity, slightly different from the previous one,  $h_0 = \|\Delta x_0\|_X \beta \gamma < 2$  and a convergence ball round  $x_0$  with radius  $\rho \sim 1/\beta \gamma$ . However, such a quantity is certainly difficult to compute in realistic nonlinear systems, if not hopeless [17]. Therefore, it's the very introduction of affine invariance framework that helps eliminate the gap between theoretical convergence analysis and realistic algorithm design and implementation.

Firstly, the contraction factor, or convergence monitor would be introduced as

$$\Theta_k = \frac{\|\Delta \theta^{k+1}\|}{\|\Delta \theta^k\|} \quad (44)$$

whenever  $\Theta_k \geq 1$  for simplicity, the algorithm monitored is classified as 'not convergent'.

Since the Broyden based method could be classified as Jacobian rank-1 update, then the Jacobian rank-1 update operator is defined as

$$E_k(J) = I - J_k J^{-1} \quad (45)$$

*Lemma 1.* For  $0 < \Theta < 1$ ,  $0 \leq \eta_0 < \Theta$  and

$$b \leq \frac{\Theta - \eta_0}{1 + \eta_0 + \frac{4}{3}(1 - \Theta)^{-1}} \quad (46)$$

if

$$\eta = \eta_0 + \frac{t}{(1 - b)(1 - \Theta)}$$

we could have

$$\eta + (1 + \eta)t < \Theta \quad (47)$$

The proof of Lemma 1 could be accessed at [17].

*Theorem 1.* For  $F \in C^1(D)$ ,  $F : D \subset \mathbb{R}^n \rightarrow \mathbb{R}^m$ ,  $D$  convex, if  $\theta^*$  indicates the unique solution of robot joint value with  $F'(\theta^*)$  nonsingular. For a specified  $\omega < \infty$ ,  $\theta \in D$ , the affine contravariance Lipschitz condition

$$\|(F'(\theta_k) - F'(\theta^*))(\theta - \theta_k)\| \leq \omega \|F'(\theta^*)(\theta_k - \theta^*)\| \|F'(\theta^*)(\theta - \theta_k)\| \quad (48)$$

holds. For specified  $\Theta \in (0, 1)$ , if  $\bar{\eta}_0 = \|E_0\| < \bar{\Theta}$ , and the initial robot joint value condition

$$b = \omega \|F'(\theta^*)(\theta_0 - \theta^*)\| \leq \frac{\bar{\Theta} - \bar{\eta}_0}{1 + \bar{\eta}_0 + 4/3(1 - \bar{\Theta})^{-1}} \quad (49)$$

holds, the robot joint value series  $\{\theta_k\}$  would converge to  $\theta^*$  in terms of error as

$$\|F_{k+1}\| \leq \bar{\Theta} \|F_k\| \quad (50)$$

or in the terms of image feature vector as

$$\|y_{k+1} - y^*\| \leq \bar{\Theta}_k \|y_k - y^*\| \quad (51)$$

*Proof.* The Jacobian rank-1 update of  $F$  is

$$\begin{aligned} F_{k+1} &= F_k + \int_{q=0}^1 F'(\theta_k + q\Delta\theta_k)\Delta\theta_k dq \\ &= \int_{q=0}^1 (F'(\theta_k + q\Delta\theta_k) - F'(\theta^*))\Delta\theta_k dq \\ &\quad + (F'(\theta^*) - J_k)\Delta\theta_k \end{aligned} \quad (52)$$

Consider the Lipschitz condition (48), (52) yields

$$\begin{aligned} \|F_{k+1}\| &\leq \int_{q=0}^1 \|(F'(\theta_k + q\Delta\theta_k) - F'(\theta^*))\Delta\theta_k\| dq \\ &\quad + \|(F'(\theta^*)J_k^{-1} - I)F_k\| \\ &\leq \int_{q=0}^1 \omega \|F'(\theta^*)(\theta_k + q\Delta\theta_k - \theta^*)\| \\ &\quad \cdot \|F'(\theta^*)\Delta\theta_k\| dq + \|E_k F_k\| \\ &\leq \int_{q=0}^1 \omega (\|F'(\theta^*)(1 - q)(\theta_k - \theta^*)\| \\ &\quad + \|F'(\theta^*)q(\theta_{k+1} - \theta^*)\|) \\ &\quad \|F'(\theta^*)\Delta\theta_k\| dq + \bar{\eta}_k \|F_k\| \\ &= \frac{1}{2}(b_k + b_{k+1}) \|F'(\theta^*)\Delta\theta_k\| + \bar{\eta}_k \|F_k\| \end{aligned} \quad (53)$$

Defining  $\bar{b}_k = \frac{1}{2}(b_k + b_{k+1})$ , then

$$\begin{aligned} \|F_{k+1}\| &\leq \bar{b}_k \|(E_k - I)F_k\| + \bar{\eta}_k \|F_k\| \\ &\leq (\bar{b}_k(1 + \bar{\eta}_k) + \bar{\eta}_k) \|F_k\| \end{aligned} \quad (54)$$

Now we only have to prove

$$\bar{b}_k(1 + \bar{\eta}_k) + \bar{\eta}_k \leq \|\bar{\Theta}\| \quad (55)$$

The update formula for  $g_k$  is

$$\begin{aligned} g_{k+1} &= g_k - F'(\theta^*)J_k^{-1}F_k = F'(\theta^*)(\theta_k - \theta^*) \\ &\quad - F_k + E_k F_k \\ &= \int_{s=0}^1 (F'(\theta^*) - F'(\theta^* + q(\theta_k - \theta^*))) \\ &\quad \cdot (\theta_k - \theta^*) dq + E_k F_k \end{aligned} \quad (56)$$

and

$$\begin{aligned} \|g_{k+1}\| &\leq \int_{s=0}^1 q\omega \|F'(\theta^*)(\theta_k - \theta^*)\| \|F'(\theta^*) \\ &\quad \cdot (\theta_k - \theta^*)\| dq + \bar{\eta}_k \|F_k\| \\ &\leq \frac{\omega}{2} \|g_k\|^2 + \bar{\eta}_k (\|g_k - F_k\| + \|g_k\|) \end{aligned} \quad (57)$$

Similarly,

$$b_{k+1} \leq \frac{1}{2}b_k^2 + \bar{\eta}_k(\frac{1}{2}b_k^2 + b_k) = \left(\bar{\eta}_k + \frac{1 + \bar{\eta}_k}{2}\right) b_k \quad (58)$$

After that, the approximation properties of the Jacobian updates could be discussed by introducing the orthogonal projection

$$Q_k = \frac{\Delta F_{k+1} \Delta F_{k+1}^T}{\|\Delta F_{k+1}\|^2} \quad (59)$$

onto the secant direction  $\Delta F_{k+1}$ , and the deterioration matrix could be noted as

$$E_{k+1} = E_k Q_k^\perp + E_{k+1} Q_k \quad (60)$$

From the ‘good’ Broyden update proof [17]

$$\begin{aligned} \bar{\eta}_{k+1} &= \|E_{k+1}\| \leq \|E_k Q_k^\perp\| + \|E_{k+1} Q_k\| \\ &\leq \|E_k\| + \frac{\|E_{k+1} \Delta F_{k+1}\|}{\|\Delta F_{k+1}\|} \end{aligned} \quad (61)$$

The second right hand term could be expressed, utilizing the secant condition, as

$$\begin{aligned} E_{k+1} \Delta F_{k+1} &= \Delta F_{k+1} - F'(\theta^*)J_{k+1}^{-1} \Delta F_{k+1} \\ &= \Delta F_{k+1} - F'(\theta^*)\Delta\theta_k \\ &= \int_{q=0}^1 (F'(\theta_k + q\Delta\theta_k) - F'(\theta^*))\Delta\theta_k dq \end{aligned} \quad (62)$$

Now the norm of (62) could be noted as

$$\begin{aligned} \|E_{k+1} \Delta F_{k+1}\| &\leq \bar{b}_k \|F'(\theta^*)\Delta\theta_k\| \\ &= \bar{b}_k \|E_{k+1} \Delta F_{k+1} - \Delta F_{k+1}\| \\ &\leq \bar{b}_k (\|E_{k+1} \Delta F_{k+1}\| + \|\Delta F_{k+1}\|) \\ &\leq \frac{\bar{b}_k}{1 - \bar{b}_k} \|\Delta F_{k+1}\| \end{aligned} \quad (63)$$

Substitute (61) into (63) yields fairly rough estimation

$$\bar{\eta}_{k+1} \leq \bar{\eta} + \frac{\bar{b}_k}{1 - \bar{b}_k} \quad (64)$$

Since

$$\bar{\eta}_k \leq \bar{\eta}_0 + \frac{\sum_{i=0}^{k-1} \bar{\Theta}_i t_0}{1 - b_0} \leq \bar{\eta} \quad (65)$$

with

$$\bar{\eta} = \bar{\eta}_0 + \frac{b_0}{(1 - b_0)(1 - \bar{\Theta})} \quad (66)$$

and

$$b_k \leq \bar{\Theta}_k b_0 \quad (67)$$

and combining (67) and Lemma 1, immediately we have

$$b_{k+1} \leq (\bar{\eta} + (1 + \bar{\eta})b_0)b_k \leq \bar{\Theta} b_k \leq \bar{\Theta}_{k+1} b_0 \quad (68)$$

and therefore

$$\begin{aligned} \bar{\eta}_{k+1} &\leq \bar{\eta}_k + \frac{b_k}{1-b_0} \leq \bar{\eta}_0 + \frac{\sum_{i=0}^{k-1} \bar{\Theta}_i t_0}{1-b_0} + \frac{\bar{\Theta}_{k+1} b_0}{1-b_0} \\ &\leq \bar{\eta}_0 + \frac{\sum_{i=0}^k \bar{\Theta}_i b_0}{1-b_0} \leq \bar{\eta} \end{aligned} \quad (69)$$

When the 'bounded deterioration property'

$$\bar{\eta}_k \leq \bar{\eta} \quad (70)$$

holds, and the error contraction

$$b_{k+1} \leq b_k \quad (71)$$

for any k. Therefore in conclusion, combining (54) and again Lemma 1 would result in

$$\|F_{k+1}\| \leq \bar{\Theta} \|F_k\| \quad (72)$$

i.e.

$$\|y_{k+1} - y^*\| \leq \bar{\Theta} \|y_k - y^*\| \quad (73)$$

**Miao Hao** received his Bachelor's Degree of Science from School of Computer Science and Technology, Xidian University in 2005 and currently is a Ph.D. candidate of National Laboratory of Information Science and Technology, Department of Computer Science and Technology, Tsinghua University, China under supervision of Prof. Zengqi Sun. His research interests include Robotics, Visual Servoing, Intelligent Control and Decision Making Support.

**Peter Deuffhard** graduated from Diploma Degree in Physics, TH Munich in 1968, and obtained his Ph.D. in Mathematics in University of Cologne, 1972. He is now the founder and president of ZIB as a scientific research institute and also Full Professor FU Berlin, chair Scientific Computing from 1986. He is co-founder of DFG Research Center Matheon in 2002 and Member of the Executive Board of Matheon. His research interests include Scientific, Engineering, and Industrial Modelling Simulation, Inverse Problems, Optimization, and Visualization Numerical Analysis/Scientific Computing: Nonlinear Systems, Ordinary Differential Equations, Partial Differential Equations and Markov Chains. He's published 17 books and over 160 academic papers and won the ICIAM Maxwell prize in 2007 which dedicated only for fundamental contributors to applied mathematics.

**Zengqi Sun** graduated from Tsinghua University, China, in 1966 and received the Ph.D degree at Chalmers University of Technology, Sweden, in 1981. He is currently a professor of the Department of Computer Science and Technology, Tsinghua University, China. He is also a Vice-President of the Chinese Association of Artificial Intelligence, Standing Council Member of the Chinese Association of Automation, the Chair of RoboCup China Committee, the Chair of Intelligent Automation Society of the Chinese Association of Automation, the Chair of the Robot Competition Committee of Chinese Association of Automation, Editors for Journal of China Science, Journal of Control Theory and Applications, Journal of System Simulation, Journal of Robotics, Journal of Intelligent Systems, International Journal of Control, Automation and Systems. He is the author or co-author of over 200 papers and the author of seven books on intelligent control and robotics. His current research interests include intelligent control, robotics, fuzzy systems, neural networks and evolution computing etc.

**Masakazu Fujii** received his B.E. and M.E. degrees in engineering from Osaka University, Toyonaka, Japan in 2000 and 2002, respectively. He is currently a researcher in Control and Communication System Department of Corporate Research and Development at IHI Corporation in Japan. His research interests include the intelligent control of manipulators and mobile robots, robot architecture and optimal control theories.