Compressive Feedback based Non-vector Space Control

Jianguo Zhao, Bo Song, Ning Xi, King Wai Chiu Lai, Hongzhi Chen, and Chengeng Qu

***Abstract*— A non-vector space control method based on compressive feedback is presented in this paper. The non-vector space means the control is not performed in traditional vector space but in the space of sets. Consider an initial set and a goal set, a stabilizing controller is designed to make the set dynamics converge to the goal set. The compressive feedback means the controller works even when only partial elements of the feedback set are available; that is, the same controller can still stabilize the set dynamics around the goal set with the compressive feedback. The controller is applied to visual servoing by considering images as sets. In this way, image processing for feature extraction is not required, which is an essential process in conventional servo methods. Moreover, the compressive feedback can reduce the size of feedback image. It is important when the sampling is time consuming such as imaging using atomic force microscopy (AFM). The visual servoing formulation of the controller is further applied to the motion control in nanomanipulations. Simulation results suggest good performance of the controller. The framework proposed in this paper can be extended to other systems where the signals can be represented as sets.**

I. INTRODUCTION

The motivation of the research presented in this paper comes from finding visual servoing methods that are free of feature extraction. For traditional image based visual servoing methods, prominent features are first extracted from the image, and then a controller is designed to make the vector of feature positions converge to a desired value [1]. Two issues exist with this feature based vector control method. On the one hand, robust feature extraction and tracking are difficult in natural environment. In fact, most visual servoing experiments are based on artificial fiducial markers. On the other hand, feature extraction suffers from information loss because only the feature information is used for control.

To address these issues, we present a set based method by considering the images as sets. Since the addition and scalar multiplication cannot be properly defined, the space of sets does not have the linear structure of vector space. Therefore, we call the approach non-vector space control method. This method can be applied to the visual servoing problem if the images are considered as sets with the image pixels as the elements in the sets. Compared with the traditional image based visual servoing approaches, the non-vector space control method uses all the image information, and no feature extraction is needed. Initial results for the non-vector space controller have been reported in [2].

The control method can be extended to the case when only part of the set elements is known, i.e., incomplete feedback. We call it compressive feedback in this paper because it relies on the results from the recent compressive sensing literature [3]. With the compressive feedback, the non-vector space control can be applied to visual servoing problem when the acquisition of a complete image is time consuming. A typical example is AFM because it can only acquire images pixel by pixel [4].

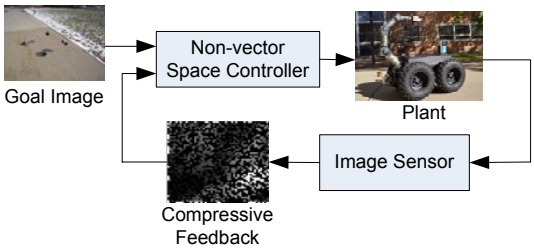

Fig. 1. Non-vector space control schematic for visual servoing

The schematic for the compressive feedback based non-vector space control method is illustrated in Fig. 1, where the visual servoing is used as an example for better explanation. First of all, a desired goal image is chosen. Based on the compressive feedback image and the goal image, the non-vector space controller will generate a signal to control the plant so that the goal image can be achieved.

The non-vector space control comes from a general framework called mutation analysis for set evolutions, which is proposed by J. P. Aubin [5]. Mutation analysis provides a natural way to describe the physical phenomenon because some objects such as shapes and images are basically sets. It has been applied to image segmentation [6], visual servoing [7], and surveillance networks [8].

Since visual servoing is employed as an application for the non-vector space control, we also provide a brief review for servoing approaches directly based on the image intensities instead of feature extraction. Recent approaches include the kernel-based method [9], the sum-of-squared-difference of intensities method [10], [11], and the entropy based method [12]. Our approach is fundamentally different from above methods because our model is formulated in non-vector space.

The rest of paper is organized as follows. First of all, the non-vector space controller is developed in section II when all the elements in the feedback set are known. In section

This work is partially supported by NSF Grant NO. IIS-0713346 and ONR Grant NOs. N00014-07-1-0935 and N00014-04-1-0799.

Jianguo Zhao, Bo Song, Ning Xi, King Wai Chiu Lai, Hongzhi Chen, and Chengeng Qu are with Department of Electrical and Computer Engineering, Michigan State University, East Lansing, MI, 48824, USA. {zhaojia1, songbo}@msu.edu, {xin, kinglai}@egr.msu.edu, {chenhon5, quchenge}@msu.edu

III, the controller is extended to the compressive feedback case when only part of the elements in the feedback set is available. Finally, the controller is applied to visual servoing for AFM based nanomanipulations, and simulation results are presented to verify the controller in section IV.

II. Non-vector Space Control based on Full Feedback

In this section, the essentials of mutation analysis will be reviewed and the controller to stabilize set dynamics around a desired set will be presented. Note that the review in this section is included for the sake of completeness, and the details can be found in books such as [5] and [13] .

A. Essentials of Mutation Analysis

To study the set dynamics, a proper metric based on Hausdorff distance should be equipped with the space of sets. The distance between a point $x \in \mathbb{R}^n$ and a set $K \subset \mathbb{R}^n$ is defined as $d_K(x) = \inf_{y\in K} ||y-x||_2$, where $||\cdot||_2$ is the Euclidian norm. The projection of x to K is defined as $P_K(x) = \{y \in K : ||y-x||_2 = d_K(x)\}$. The distance from a set $X \subset \mathbb{R}^n$ to another set $Y \subset \mathbb{R}^n$ is $d(X,Y) = \sup_{x\in X} d_Y(x)$; similarly, $d(Y,X) = \sup_{y\in Y} d_X(y)$. Note that in general, $d(X,Y) \neq d(Y,X)$. The Hausdorff distance defined by

$$dh(X,Y) = \max\{d(X,Y), d(Y,X))\} \tag{1}$$

can be considered as a metric in the space of sets. Before the dynamics of sets can be defined, a few extra definitions should be introduced. Firstly, let $\varphi : E \mapsto \mathbb{R}^n$ with $E \subset \mathbb{R}^n$ be a bounded Lipschitz function. Denote the set of all such functions as $\mathrm{BL}(E, \mathbb{R}^n)$. Secondly, a tube $K(t) \subset \mathbb{R}^n$ is a mapping: $K : \mathbb{R}^+ \mapsto 2^{\mathbb{R}^n}$, where $2^{\mathbb{R}^n}$ is the powerset of $\mathbb{R}^n$. Thirdly, the transition for $\varphi \in \mathrm{BL}(E, \mathbb{R}^n)$ is defined as

$$T_\varphi(t, K_0) = \{x(t) : \dot{x} = \varphi(x), x(0) \in K_0\} \tag{2}$$

where $K_0 \subset \mathbb{R}^n$ can be considered as the set of initial points of the ordinary differential equation $\dot{x} = \varphi(x)$. Note that $T_\varphi(t, K_0)$ is also a set of points at time t. Fourthly, the derivative of a tube is the φ that satisfies the following equation:

$$\lim_{\Delta t \to 0^+} \frac{1}{\Delta t} dh(K(t+\Delta t), T_\varphi(\Delta t, K(t))) = 0 \tag{3}$$

Fifthly, the mutation of set is defined as follows:

$$\mathring{K}(t) = \{\varphi(x) \in \mathrm{BL}(E, \mathbb{R}^n) : \text{Eq. (3) is satisfied}\} \tag{4}$$

Finally, the dynamics of sets can be described by the mutation equation as:

$$\varphi(x) \in \mathring{K}(t) \tag{5}$$

Similarly, the controlled mutation equation can be obtained by considering the map $\varphi : E \times U \mapsto \mathrm{BL}(E, \mathbb{R}^n)$ where U is the set of all possible controls u:

$$\varphi(x(t), u(t)) \in \mathring{K}(t) \quad \text{with} \quad u(t) = \gamma(K(t)) \tag{6}$$

where $\gamma : 2^{\mathbb{R}^n} \mapsto U$ is the feedback map from set $K(t)$ to the control input.

B. Stabilizing Controller based on Full Feedback

Based on mutation analysis, the stabilization problem can be stated as: Given the goal set $\hat{K}$ and an initial set $K(0)$, design a controller $u(t) = \gamma(K(t))$ based on current set $K(t)$ such that $dh(K(t), \hat{K}) \to 0$ as $t \to \infty$.

A controller can be designed if $\varphi(x(t), u(t))$ in the controlled mutation Eq. (6) is linear in u. Without loss of generality, let $\varphi(x,u) = L(x)u$ with $u \in \mathbb{R}^n$ and $L(x) \in \mathbb{R}^{m\times n}$. Note that the time t is omitted for clear presentation. To obtain the stabilizing controller, the following Lyapunov function candidate can be used:

$$V(K) = \int_K d^2_{\hat{K}}(x)dx + \int_{\hat{K}} d^2_K(x)dx \tag{7}$$

Note that $V(K)$ is different from traditional Lyapunov function in vector space because it is a function of sets, and its derivative requires special techniques. Based on this function, we have the following theorem:

Theorem 1: [2], [7] For system $L(x)u \in \mathring{K}(t)$ with $x \in \mathbb{R}^m$, $L(x) \in \mathbb{R}^{m\times n}$, $u \in \mathbb{R}^n$, and $K \subset \mathbb{R}^m$, the following controller can locally exponentially stabilize it at $\hat{K}$:

$$u(t) = \gamma(K) = -\alpha D(K)^+ V(K) \tag{8}$$

where $\alpha > 0$ is a gain factor and $D(K)^+$ is the Moore-Penrose pseudoinverse of $D(K) \in \mathbb{R}^{1\times n}$ defined by:

$$\begin{aligned} D(K) = &\int_K d^2_{\hat{K}}(x)(\sum_{i=1}^m \frac{\partial L_i}{\partial x_i})dx + 2\int_K (x - P_{\hat{K}}(x))^{\mathrm{T}} L(x)dx \\ &- 2\int_{\hat{K}} (x - P_K(x))^{\mathrm{T}} L(P_K(x))dx \end{aligned} \tag{9}$$

where $L_i\,(i = 1, 2, \ldots, m)$ is the i-th row vector in matrix L, and $\partial L_i/\partial x_i$ is also a row vector with the same dimension. Using the controller in Eq. (8), the stabilization problem at the beginning of this subsection can be solved. The proof of this theorem is given in [7] based on the extension of Lyapunov theory to the space of sets.

The controller can be applied to visual servoing problem. In fact, for servoing with grey scale images, $x = [x_1, x_2, x_3]^{\mathrm{T}}$ where x_1 and x_2 are the pixel indices, and x_3 is the pixel intensity. The control input $u(t)$ has three translational components and three rotational components; therefore, let $u(t) = [v_x, v_y, v_z, \omega_x, \omega_y, \omega_z]^{\mathrm{T}}$. Based on the perspective projection sensing model, we can obtain the following relation:

$$\dot{x}(t) = L(x(t))u(t) \tag{10}$$

where

$$L = \begin{bmatrix} -1 & 0 & x_1 & x_1x_2 & -(1+x_1^2) & x_2 \\ 0 & -1 & x_2 & 1+x_2^2 & -x_1x_2 & -x_1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Note that above matrix is obtained from the interaction matrix in visual servoing by assuming the depth and focal length to be one. From Eq. (10), $\varphi(x(t), u(t)) = L(x(t))u(t)$ is linear in u; therefore, the controller in Eq. (8) can be applied.

The controller, however, only deals with the case when the full current set $K(t)$ is available. In some cases such as AFM based nanomanipulations, it will be time consuming to obtain the full image set $K(t)$. In this case, only part of $K(t)$ will be

sampled to increase the sampling rate. The reduced sampled set will be termed as compressive feedback because the idea comes from the recent compressive sensing literature.

III. Non-vector Space Control based on Compressive Feedback

If only a subset of the whole set is used for feedback, then the stabilization problem can be restated as: Given the goal reduced sampled set $\hat{K}_c \subset \hat{K}$ and an initial reduced sampled set $K_c(0) \subset K(0)$, design a controller $u(t) = \gamma(K_c(t))$ based on current reduced sampled set $K_c(t) \subset K(t)$ (compressive feedback) such that $dh(K(t), \hat{K}) \to 0$ as $t \to \infty$.

In this section, we will show that under certain conditions, the stabilization can be accomplished with compressive feedback using the same controller in Eq. (8). The essentials of compressive sensing will be reviewed and relevant results will be presented for the stability analysis.

A. Compressive Sensing for Feedback

Compressive sensing is a general framework for acquiring sparse or approximately sparse signals. It has been successfully applied to pattern recognition [14] and control [15], [16]. One of the major results in compressive sensing is that if a high dimensional signal is sparse or sparse in some basis, then the signal can be recovered from certain low dimensional linear observations. Therefore, with the reduced sampled set $K_c(t)$, if we can recover the original set $K(t)$, the controller in Eq. (8) can still be used, but the sampling time can be reduced such as in the case of AFM sampling. Note that additional time for recovery is still needed.

For an unknown sparse signal $x \in \mathbb{R}^n$, the general compressive sensing problem can be summarized as:

$$y = Ax \tag{11}$$

where $y \in \mathbb{R}^m$ is the available linear measurements and $A \in \mathbb{R}^{m \times n}$ is the sensing matrix mapping the original high dimension signal x to a low dimensional measurement y.

With $m < n$, Eq. (11) is underdetermined and infinite many solutions exist. In compressive sensing, the problem is formulated as an optimization problem to minimize the number of nonzero elements in x. In this case, x can be successfully recovered from y provided that the sensing matrix A satisfies specific property to insure that the essential information in x is retained in y. The most popular property is the restricted isometry property.

Definition 1: [17] A matrix $A \in \mathbb{R}^{m \times n}$ satisfies the restricted isometry property (RIP) of order S if there exists a constant $\delta_S \in (0,1)$ such that:

$$(1-\delta_S)||x||_2^2 \leq ||Ax||_2^2 \leq (1+\delta_S)||x||_2^2 \tag{12}$$

for all the S-sparse vectors $x \in \mathbb{R}^n$ where an S-sparse vector means it has at most S nonzero elements.

The verification of a matrix satisfying RIP has combinatorial computation complexity. But random matrices such as Gaussian matrix, binary matrix, or randomly sampled Fourier matrix have shown to satisfy the RIP with very high probability [17]. Therefore, a sparse signal in the time domain can be reconstructed from a small set of random Fourier coefficients [18]. Similarly, a sparse signal in the frequency domain can be reconstructed with a small set of random samples in the time domain.

For our problem where only a subset of the full set is sampled, sensing matrix can be designed to satisfy the RIP. Take the image set for example, let the signal $x \in \mathbb{R}^n$ be obtained from the image by stacking the intensity values row by row. Moreover, assume it is sparse in the frequency domain, i.e., the coefficient vector $x' \in \mathbb{R}^n$ after the discrete Fourier transform (DFT) is sparse. Let the DFT matrix be $\Psi \in \mathbb{R}^{n \times n}$, then $x' = \Psi x$. Moreover, let $\Phi \in \mathbb{R}^{m \times n}$ be a matrix formed by uniformly selecting m rows at random from the n-dimensional standard basis. Then if $y = \Phi x = \Phi \Psi^{-1} x'$ where y is the measurements, this corresponds to uniformly sample the pixel intensities at random. The recoverability of x' from y is guaranteed by the following lemma:

Lemma 1: [19] Let Φ and Ψ be as defined before, then matrix $A = \Phi\Psi^{-1}$ satisfy the RIP of order S with very high probability if

$$m \geq C \cdot S \cdot (\log n)^4 \tag{13}$$

where C is a constant independent of m and n.

The matrix A in the above lemma can be considered as the sensing matrix for sparse vector x'. Therefore, if x' is S-sparse, it can be recovered from y, and the original signal can be obtained from $x = \Psi^{-1}x'$.

Once the original signal can be recovered, the stabilization problem is the same as discussed in previous section. The signal recovery, however, still needs extra time especially when the signal's dimension is high. Therefore, it is better to directly use the compressive feedback without the recovery process, which is discussed in the beginning of this section. In this case, further investigations should be performed.

B. Stabilizing Controller based on Compressive Feedback

With compressive feedback, the mutation equation is still the same. Since both $K_c(t) \subset K(t)$ and $\hat{K}_c \subset \hat{K}$ are still sets, the same controller can be used to locally exponentially stabilize K_c at $\hat{K}_c$:

$$u(t) = \gamma(K_c) = -\alpha D(K_c)^+ V(K_c) \tag{14}$$

where $\alpha > 0$ is a gain factor, $D(K_c)^+$ is the pseudoinverse of $D(K_c) \in \mathbb{R}^{1 \times n}$ defined by replacing K with K_c and $\hat{K}$ with $\hat{K}_c$ in Eq. (9), and $V(K_c)$ is obtained similarly.

With the above controller, we have $dh(K_c(t), \hat{K}_c) \to 0$ as $t \to \infty$. But our goal of $dh(K(t), \hat{K}) \to 0$ may not be achieved. It is possible that there exists another set $\widetilde{K}$ with $\hat{K}_c \subset \widetilde{K}$. This can happen if the cardinality of $\hat{K}_c$ is much less than that of $\hat{K}$. In this case, $K_c(t)$ may actually converge to $\widetilde{K}$ instead of $\hat{K}$. To avoid such circumstances, we need to show that if $dh(K_c, \hat{K}_c) \to 0$, then $dh(K, \hat{K}) \to 0$ under certain conditions.

In fact, with the conditions from compressive sensing, we can show that $dh(K_c, \hat{K}_c) \to 0 \Rightarrow dh(K, \hat{K}) \to 0$. Still take the image as example. Let x_k be the vector of all the intensities of image set K. Moreover, let any element in K be a three dimensional vector with the first two as the pixel indices

and the third as the intensity. Without loss of generality, let the elements of set K be obtained in the same order as the intensity vector x_k is formed. Similarly, let $\hat{x}_k$, x_{k_c}, and $\hat{x}_{k_c}$ be the vector of all the intensities of image sets $\hat{K}$, K_c, and $\hat{K}_c$, respectively. Then the following lemma can be established.

Lemma 2: $dh(K,\hat{K}) \to 0$ if and only if $||x_k - \hat{x}_k||_2 \to 0$

Proof: (1) First of all, let's show $dh(K,\hat{K}) \to 0 \Rightarrow ||x_k - \hat{x}_k||_2 \to 0$. By the definition of Hausdorff distance, if $dh(K,\hat{K}) \to 0$, then for any $p = [p_1, p_2, p_3]^{\mathrm{T}}$ in set K, we have $\min_{q\in\hat{K}} ||p-q||_2 \to 0$. Let $q = [q_1, q_2, q_3]^{\mathrm{T}}$ be the element in $\hat{K}$ when the minimum is achieved, then $(p_1-q_1)^2 + (p_2-q_2)^2 + (p_3-q_3)^2 \to 0$. Because the first two coordinates in p and q are the pixel indices, $(p_1 - q_1)^2 + (p_2-q_2)^2$ cannot approach zero if the indices are different. Therefore, $p_1 = q_1$ and $p_2 = q_2$, which means p and q are at the same location of K and $\hat{K}$. Moreover, we have $(p_3-q_3)^2 \to 0$. Consequently, we have $||x_k - \hat{x}_k||_2 \to 0$ since $||x_k - \hat{x}_k||_2^2$ is the sum of all the square of intensity differences for the same pixel indices such as $(p_3-q_3)^2$.

(2) Second, let's show $||x_k - \hat{x}_k||_2 \to 0 \Rightarrow dh(K,\hat{K}) \to 0$. Let $p = [p_1, p_2, p_3]^{\mathrm{T}} \in K$ and $q = [q_1, q_2, q_3]^{\mathrm{T}} \in \hat{K}$ be two arbitrary elements with the same pixel indices, i.e., $p_1 = q_1$ and $p_2 = q_2$. Since $||x_k - \hat{x}_k||_2 \to 0$, we have $(p_3-q_3)^2 \to 0$. Then for $p \in K$, we have $\min_{q'\in\hat{K}} ||p-q'||_2 \le ||p-q||_2 \to 0$. For any other elements in K, we also have similar arguments. Therefore, $\max_{p'\in K} \min_{q'\in\hat{K}} ||p'-q'||_2 \to 0$. Similarly, we have $\max_{q'\in\hat{K}} \min_{p'\in K} ||q'-p'||_2 \to 0$. Therefore, $dh(K,\hat{K}) \to 0$ by the definition of Hausdorff distance. ■

Although the above arguments are based on images, they can be applied to other signals. In fact, any signal can be considered as a set with each element a two dimensional vector, where the two numbers in the vector represent a signal value and its corresponding location in the signal. Based on previous two lemmas, we have the following proposition:

Propsition 1: Assume $x_k \in \mathbb{R}^n$ and $\hat{x}_k \in \mathbb{R}^n$ be S-sparse in the frequency domain, $x_{k_c} \in \mathbb{R}^m$ and $\hat{x}_{k_c} \in \mathbb{R}^m$ be obtained uniformly at random from x_k and $\hat{x}_k$, respectively. If

$$m \ge 2 \cdot C \cdot S \cdot (\log n)^4$$

where C is a constant. Then with high probability, we have $dh(K,\hat{K}) \to 0$ if $dh(K_c,\hat{K}_c) \to 0$.

Proof: With random sampling, we have $x_{k_c} = \Phi x_k = \Phi\Psi^{-1}x'_k = Ax'_k$ where Φ, Ψ, and A are defined as in lemma 1, x'_k is the DFT coefficients for x_k. Similarly, we have $\hat{x}_{k_c} = \Phi\hat{x}_k = A\hat{x}'_k$. Since $m \ge 2\cdot C\cdot S\cdot(\log n)^4$, $A \in \mathbb{R}^{m\times n}$ satisfies the RIP with order $2S$ from lemma 1. Let the RIP constant be δ_{2S}. Since $x_k \in \mathbb{R}^n$ and $\hat{x}_k \in \mathbb{R}^n$ are S-sparse in the frequency domain, x'_k and $\hat{x}'_k$ are S-sparse signals; therefore, $x'_k - \hat{x}'_k$ is $2S$-sparse.

Using lemma 2, we have $||x_{k_c} - \hat{x}_{k_c}||_2 \to 0$ from $dh(K_c,\hat{K}_c) \to 0$. Form the RIP property, we have $||x_{k_c} - \hat{x}_{k_c}||_2^2 = ||Ax'_k - A\hat{x}'_k||_2^2 = ||A(x'_k - \hat{x}'_k)||_2^2 \ge (1-\delta_{2S})||x'_k - \hat{x}'_k||_2^2$. Since $1-\delta_{2S} > 0$, we have $||x'_k - \hat{x}'_k||_2 \to 0$. Since the DFT matrix Ψ is orthonormal,

$$||x'_k - \hat{x}'_k||_2 = ||\Psi^{-1}(x'_k - \hat{x}'_k)||_2 = ||x_k - \hat{x}_k||_2$$

Therefore, $||x_k - \hat{x}_k||_2 \to 0$. Based on lemma 2 again, we have $dh(K,\hat{K}) \to 0$. ■

This proposition suggests that, under certain conditions, the controller in Eq. (14) can be employed for the stabilization problem proposed at the beginning of this section.

IV. Applications and Testing

To validate the proposed method, we choose the applications of AFM based nanomanipulations. Originally designed as an imaging tool at the nano-scale, AFM has widely been applied to manipulate various nano-objects by using a sharp tip as a nano-scale robotic manipulator [4]. Examples include cell analysis [20], nanosensor fabrication [21], fast imaging on dynamic change samples [22], etc.

Despite the wide applications of AFM based nanomanipulations, many problems still exist. One of the major problems is the inaccurate position control. Since nanomanipulation is achieved by moving the AFM tip from one position to a desired position, it relies on accurate point-to-point position control. Currently, piezoelectric tubes are widely used to actuate AFM's probe movement. Nevertheless, the inherent nonlinearities of piezoelectric actuators such as hysteresis, creep, vibration, and thermal drift make high precision position control extremely difficult [23].

Two methods are frequently used to obtain accurate position control. The first one relies on position sensors to perform feedback control. The sensors, however, can only be added to the piezo tube instead of AFM's tip. Moreover, sensor noise in the nanoscale makes them incapable to provide accurate feedbacks. This is why for high performance AFM, the open loop scanner is still used to achieve atom level resolution (such as the Multimode AFM, Bruker Inc.). The second method tries to eliminate all the adverse effects by a feedforward compensation based on model inversion [24]; however, this method requires model identification process and the model may also be time varying.

The non-vector space control method provides another approach for precise position control. The underlying idea is to use AFM image as feedback to realize closed-loop control. In fact, the AFM tip can be considered as a single pixel camera with two degree of freedoms. By moving the AFM tip locally in an area, a current local scan image can be obtained [25]. If a desired local scan image around a desired tip position is given, then the non-vector space controller can be used to steer the current local scan image to the desired image. Consequently, the tip position can also be steered to the desired position.

A. Application to Visual Servoing for Nanomanipulation

The two cases with full and compressive feedback can be implemented using the controllers in Eq. (8) and Eq. (14), respectively. AFM tip movement is a special case with only two translational velocity components v_x and v_y as the inputs. Therefore, $\varphi(x,u)$ is obtained from Eq. (10) as:

$$\varphi(x,u) = Lu$$

where

$$L = \begin{bmatrix} -1 & 0 \\ 0 & -1 \\ 0 & 0 \end{bmatrix}$$

is a constant matrix. Therefore, $D(K)$ in the controller given by Eq. (9) is simplified to:

$$D(K) = 2\{\int_K [x - P_{\hat{K}}(x)]^{\mathrm{T}} dx - \int_{\hat{K}} [x - P_K(x)]^{\mathrm{T}} dx\} L$$

The controller based on compressive feedback can be derived similarly by replacing K with K_c and $\hat{K}$ with $\hat{K}_c$.

The basic validation strategy for full feedback case is described as follows [2]. First of all, a large area of interest is scanned and an image I is obtained. Within image I, a small patch is chosen as the desired image $\hat{K}$. Then the AFM tip performs a local scan around its current tip position to obtain the current image $K(t)$. Based on the two image sets, a control input u is calculated to drive the AFM tip to an updated position. Repeat this process by updating the current image once the AFM tip reaches a new position. Eventually, the desired image can be achieved, and the AFM tip can be steered to the desired position. Note that although $u = [v_x, v_y]^{\mathrm{T}}$ is the velocity, it can be considered as displacement if the time interval for each iteration is the same.

The compressive feedback case is similar. Instead of obtaining a full local scan image around the tip's position, random samples are obtained based on a random sampling matrix. Note that random sampling for AFM may be difficult because of the special working principle of AFM, which uses a sharp tip to scan line by line on top of sample surface. But a continuous trajectory can be generated to pass all the random points and only visit each point once. With this trajectory, the sampling time can be significantly less than the full local scan if the number of random samples is small.

As our first step, we test our controller using simulations on AFM images. An entire image is used as the image I discussed above. Then, K and $\hat{K}$, sufficiently close to each other, are chosen from I. Based on $\hat{K}$ and K, u is calculated to update the position. With the updated position, a new K can be obtained, and a new u can be derived. This process will repeat until $dh(K, \hat{K}) = 0$. The simulation procedure for compressive feedback is similar to the full feedback case except that only a uniformly randomly sampled subset of both goal and current images are used.

The simulation is performed based on a DNA image with a scan size of $6.8\,\mu m$ and image size 256×256 pixels as shown in Fig. 2. The initial and goal images, with image size 30×30 pixels, are labeled and enlarged on the right side of the entire AFM image. The pixel distance between two images is eight in both horizontal and vertical directions (corresponding to 212.5 nm in real environment).

B. Testing Results based on Full Feedback

The simulation results for full feedback case are shown in Fig. 3, where the Lyapunov function values and the Hausdorff distances with respect to the iterations are plotted.

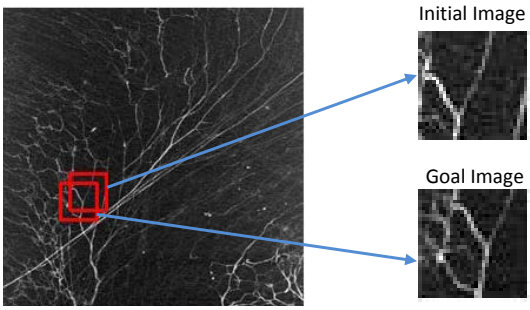

Fig. 2. Simulation images for the non-vector space control

From the figure, both the Lyapunov function value and Hausdorff distance converge to zero after five iterations, which means the control goal is achieved.

Note that the Hausdorff distance increases at the fourth iteration. This is because the decrease of Lyapunov function cannot guarantee the decrease of Hausdorff distance. But once the Lyapunov function value goes to zero, the Hausdorff distance must also approach zero [2].

The results for the full feedback case have been reported in [2], but we still include the results here for comparison with the compressive feedback case. In addition, a different simulation image is used in our previous paper.

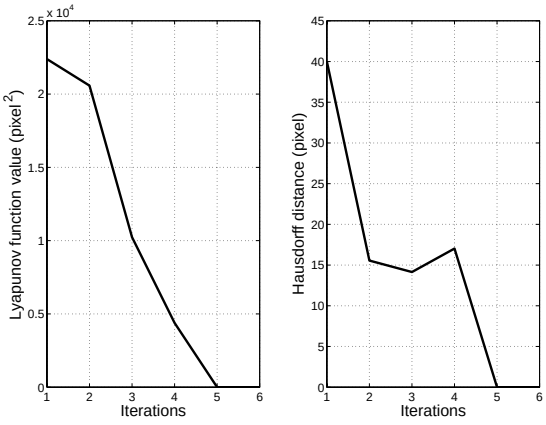

Fig. 3. Simulation results of non-vector space control with full feedback

C. Testing Results based on Compressive Feedback

For this simulation, the uniformly randomly sampled compressive feedback contains 500 pixels, which is almost half of the full feedback case with 900 pixels. Note that the number of random samples depends on the sparsity level as shown in Eq. (13), but in real applications, we cannot get the sparsity without obtaining the entire local image and performing a DFT. Therefore, we choose the number 500 empirically.

In order to make a comparison with the full feedback case, we use the same initial and goal positions in previous simulation shown in Fig. 2. Since the random samples are different for each time, there can be many different results. One of the results is shown in Fig. 4, where the number of iterations is only four. Furthermore, the computation time for the controller per iteration using the compressive feedback

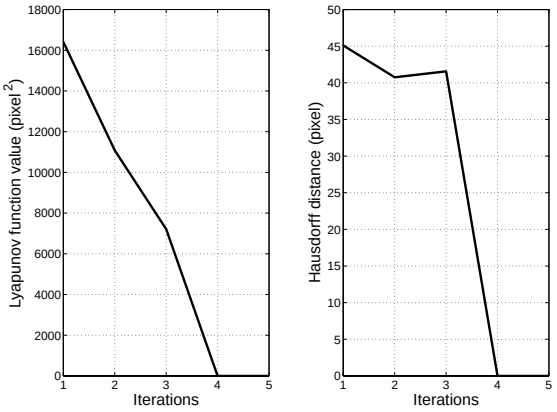

Fig. 4. Simulation results of non-vector space control with compressive feedback

(0.88 s) is much smaller than the full feedback one (2.86 s). This result shows that the non-vector space controller based on compressive feedback can achieve the same objective as the full feedback with reduced computation time. In real implementations, more time reduction can be expected since only a subset of the entire local image is needed to be sampled.

Proposition 1 suggests that the controller only works with high probability. To investigate this problem, 50 simulations are conducted to examine the convergence. Among the 50 simulations, only three of them cannot converge to the desired image. After changing the gain value α in the controller, however, these three exceptions can also converge. Therefore, the performance for the compressive feedback based controller is promising.

With this controller, the accuracy can also be quite high. In fact, the accuracy depends on the scan size of AFM image. Consider a 512×512 pixels AFM image with the scan size of $1\,\mu m$, the accuracy of this non-vector space control method will reach at least 2 nm according to the simulation.

V. Conclusions

This paper presents a non-vector space control method based on the dynamics of sets. It is shown that a controller can be designed when the full set is available for feedback. Based on ideas from compressive sensing, the controller is extended to the case when only part of the full set is available for feedback. This case, called compressive feedback in this paper, can be useful when the feedback samples are difficult to obtain or when it takes a long time to obtain the samples. The non-vector space controller is applied to visual servoing problem by considering images as sets. With this approach, there is no need to extract features in the image, and all the information in the image can be used. The controller is validated using simulations on AFM images. The results show that the compressive feedback can have the same performance as the full feedback case with almost only half of the feedback samples. The approach presented in this paper can be applied to AFM based nanomanipulations to achieve real time high precision motion control.

References

[1] F. Chaumette and S. Hutchinson, "Visual servo control part I: Basic approaches," *IEEE Robot. Autom. Mag.*, vol. 13, no. 4, pp. 82–90, 2006.

[2] J. Zhao, B. Song, N. Xi, and K. W. C. Lai, "Mutation analysis models for visual servoing in nanomanipulations," in *Proc. IEEE Int. Conf. Decision and Control*, Orlando, Florida, USA, 2011, pp. 5683–5688.

[3] E. Candès and M. Wakin, "An introduction to compressive sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21–30, 2008.

[4] S. M. Salapaka and M. V. Salapaka, "Scanning probe microscopy," *IEEE Control System Magzine*, vol. 28, no. 2, pp. 65–83, 2008.

[5] J. P. Aubin, *Mutational and Morphological Analysis: Tools for Shape Evolution and Morphogenesis*. Birkhäuser, 1998.

[6] T. Lorenz, "Set-valued maps for image segmentation," *Comput Visual Sci*, vol. 4, no. 1, pp. 41–57, 2001.

[7] L. Doyen, "Mutational equations for shapes and vision-based control," *J. Math. Imaging Vis.*, vol. 5, no. 2, pp. 99–109, 1995.

[8] A. Goradia, N. Xi, Z. Cen, and M. Mutka, "Modeling and design of mobile surveillance networks using a mutational analysis approach," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Alberta, Canada, 2005, pp. 3003–3008.

[9] V. Kallem, M. Dewan, J. P. Swensen, G. D. Hager, and N. J. Cowan, "Kernel-based visual servoing," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, San Diego, CA, USA, 2007, pp. 1975–1980.

[10] C. Collewet and E. Marchand, "Photometric visual servoing," *IEEE Transactions on Robotics*, vol. 27, no. 4, pp. 828–834, 2011.

[11] S. Han, A. Censi, A. D. Straw, and R. M. Murray, "A bio-plausible design for visual pose stabilization," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Taipei, Taiwan, 2010, pp. 5679–5686.

[12] A. Dame and E. Marchand, "Mutual information-based visual servoing," *IEEE Transactions on Robotics*, vol. 27, no. 5, pp. 1–12, 2011.

[13] T. Lorenz, *Mutation Analysis: A Joint Framework for Cauchy Problems in and Beyond Vector Spaces*. Springer, 2010.

[14] D. Hsu, S. M. Kakade, J. Langford, and T. Zhang, "Multi-label prediction via compressed sensing," in *Advances in Neural Information Processing Systems*, Vancouver, Canada, 2009.

[15] O. N. Granichin and D. V. Pavlenko, "Randomization of data acquisition and l1-optimization (recognition with compression)," *Automation and Remote Control*, vol. 71, no. 11, pp. 2259–2282, 2010.

[16] M. Wakin, B. Sanandaji, and T. Vincent, "On the observability of linear systems from random, compressive measurements," in *IEEE Conf. on Decision and Control*, Atlanta, Georgia, 2010, pp. 4447–4454.

[17] E. Candès and T. Tao, "Near optimal signal recovery from random projections: universal encoding strategies?" *IEEE Trans. Inform. Theory*, vol. 52, no. 12, pp. 5406–5425, 2006.

[18] E. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inform. Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[19] M. Rudelson and R. Vershynin, "On sparse reconstruction from Fourier and Gaussian measurements," *Commun. Pure Appl. Math.*, vol. 61, no. 8, pp. 1025–1045, 2008.

[20] C. K. M. Fung, N. Xi, R. Yang, K. Seiffert-Sinha, K. Lai, and A. A. Sinha, "Quantitative analysis of human keratinocyte cell elasticity using atomic force microscopy (afm)," *IEEE Trans. on NanoBioscience*, vol. 10, no. 1, pp. 9–15, 2008.

[21] H. Chen, N. Xi, K. Lai, C. Fung, and R. Yang, "Development of infrared detectors using single carbon-nanotube-based field-effect transistors," *IEEE Trans. on Nanotechnology*, vol. 9, no. 5, pp. 582 –589, 2010.

[22] B. Song, N. Xi, R. Yang, K. W. C. Lai, and C. Qu, "Video rate atomic force microscopy imaging," in *Proc. ANS EPRRSD - 13th Robotics and remote Systems for Hazardous Environments and 11th Emergency Preparedness and Response*, Knoxville, TN, USA, 2011.

[23] D. Croft, G. Shed, and S. Devasia, "Creep, hysteresis, and vibration compensation for piezoactuators: Atomic force microscopy application," *ASME Journal of Dynamic Systems, Measurement, and Control*, vol. 123, no. 1, pp. 35–43, 2001.

[24] K. Leang, Q. Zou, and S. Devasia, "Feedforward control of piezoactuators in atomic force microscope systems," *IEEE Control System Magazine*, vol. 29, no. 1, pp. 70–82, 2009.

[25] L. Liu, Y. Luo, N. Xi, Y. Wang, J. Zhang, and G. Li, "Sensor referenced real-time videolization of atomic force microscopy for nanomanipulations," *IEEE/ASME Trans. Mechatronics*, vol. 13, no. 1, pp. 76–85, 2008.