Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

Featureless Visual Tracking Based on Non-vector Space Control Theory ⋆

Hailin Huang * **Jianguo Zhao** ** **Ning Xi** ***

* *City University of Hongkong, Tat Chee Avenue, Kowloon, Hong Kong, 999077 (Tel: 852-2442-4622; e-mail: haihuang@cityu.edh.hk).*
** *Department of Electrical and Computer Engineering, Michigan State University, East Lansing, MI, 48824, USA (e-mail: jguozhao@gmail.com)*
*** *City University of Hongkong, Tat Chee Avenue, Kowloon, Hong Kong, 999077 (e-mail: xin@egr.msu.edu)*

Abstract: This paper proposes a featureless visual target tracking approach based on non-vector space control theory. By considering the image as a set with pixels as its elements, the visual tracking problem could be treated as the mutation control between an initial image set and a prescribed target image set, then the motion of the robot can be reflected in the dynamic of the image set. Based on mutation analysis over sets, a shape functional describing the difference between two dynamic sets is defined, the directional derivative of this shape functional is derived and a Lyapunov function is constructed to design a controller to make an initial image set to track a moving goal image set, thereby steering the robot to follow the motion of the target. A 2-dimensional translation motion case is employed as an example to illustrate the feasibility of the proposed approach.

Keywords: Mutation analysis, visual tracking, image based control, shape directional derivative.

1. INTRODUCTION

Visual servoing control in robotics is to the employ the vision sensing to move the mechanical system from an initial configuration to a desired configuration such that the image will converge to a target image. When the target is moving in a prescribed or pre-unknown pattern, the visual servoing control of the robot becomes a visual tracking problem. Visual servoing control always needs complicated requires detections and matching of geometric features in the images. However, feature detection and matching of these features are always has very low correctness rate in lots of practical environment (E. Marchand and F. Chaumette (2005)).

To address this problem, the direct visual servoing approach has become a popular research topic in recent years: Deguchi used principle component analysis to get the eigenspace with lower dimension; however, lower dimensional eigenspace should be calculated first (K. Deguchi (2000)). Kallem et al. used projective approach to get low dimensional kernel measurement, then the visual servoing problem was to minimize the difference between the initial and goal kernel measurements (Vinutha Kallem et al (2007)). Collewet et al derived the vision jacobian matrix and aimed at minimizing the sum of square differences (SSD) (Christophe Collewet and Eric Marchand (2011)). Besides the above methods, we can also use the mutual information between the target image and an initial image (Amaury Dame and Eric Marchand (2011)).

Doyen has presented a direct visual servoing control approach without any feature extraction (Luc Doyen (1995)). By considering the images as sets, a controller to drive an initial image to a target image can be obtained. Because the controller is designed to control the movement of a set rather than a single point, we can not use the difference of two vectors to describe the control process, then we call it non-vector space control theory. Continued by Doyen's work, Zhao et al have successfully applied this non-vector approach for the control of the robot to align to a prescribed target image (J. Zhao et al (2012)); however, their applications are based on the fact that the target is static and therefore not applicable to the tracking problem. In this paper, we aims at designing a controller for visual tracking of the moving target based on the non-vector space theory.

This paper is organized as follows: In section II, mathematical tools for describing the difference between sets and the dynamics of sets are given. In section III, a controller for online target tracking is designed based on the non-vector space control theory and the controller is designed under Lyapunov function so that the stability of the controller is proved. The controller is applied in a 2-dimensional translation case to show the feasibility of the proposed approach in section IV and a conclusion of this paper is drawn in the last section.

2. DYNAMICS IN NON-VECTOR SPACE

The motion of a mechanical system can be modeled as a differential equation in the vector space. For visual servoing control, if the image captured from camera is considered as a set, then the movement of the camera leads to the set evolution. The set evolution with respect to time can be considered as the tube over the space of image sets. We can not use the linear structure to describe the evolution dynamics of image set, because in this case, the state of the image can not be represented by a single vector but a collection of vectors, such that the difference between any two states of the controller can not be simply described as the difference of two vectors.

⋆ This research work is partially supported under project "Development of CityU Research and Education Center for Ultra-Automation" in City University of Hongkong.

Mathematical tools for the mutational analysis can be employed to model the dynamic of the image set without the vector structure (Jean Pierre Aubin (1998)).

2.1 Difference between sets

In this paper, we are assumed that each element x in the set X is a vector that has the same dimension, i.e., a set X is a class of vector elements. If each $x \in X$ is a n dimensional vector in real space, we call X a $\mathbb{R}^n$ set. In order to describe the different between two sets X and Y, a metric should first be defined. The Hausdorff distance is an ideal candidate to measure such difference:

$$\mathcal{D}(X,Y) = \max\{\max_{x\in X}\min_{y\in Y}\|y-x\|, \max_{y\in Y}\min_{x\in X}\|y-x\|\} \quad (1)$$

where $x \in X, y \in Y$ are the elements of the two respective sets; $\|\cdot\|$ is the Euclidean distance, the distance from a point $x \in X$ to a set Y is defined as: $d_Y(x) = \inf_{y\in Y}\|y-x\|$ and vice versa.

The projection from a point $x \in \mathbb{R}^n$ to a set $Y \subset \mathbb{R}^n$ is defined as all the points in set Y that are closest to x:

$$\Pi_Y(x) = \{y \in Y : \|y-x\| = d_Y(x)\} \quad (2)$$

If the feature vectors of dimension n are served as the set elements, then the set becomes a $\mathbb{R}^n$ set and the process of finding the projection point can be seen as the correspondence process (or feature tracking process) in the typical visual servoing problems.

2.2 Dynamic modeling for sets

The changing of each element x in a set with respect to time t could be modeled by a function of t, which is always denoted as $x(t)$. For a set of elements, however, the elements' evolution with respect to time can be modeled by a tube $K(t) \subset \mathbb{R}^n$ that is defined as: $K(t) : \mathbb{R}_{\geq 0} \mapsto \mathcal{K}(\mathbb{R}^n)$, where $\mathbb{R}_{\geq 0}$ is the non-negative real numbers and $\mathcal{K}(\mathbb{R}^n)$ is the set of all nonempty compact subsets of $\mathbb{R}^n$. Let $\varphi : E \mapsto \mathbb{R}^n$ with $E \subset \mathbb{R}^n$ be a bounded Lipschitz function, and the set of all this type of functions is given as as $\mathrm{BL}(E,\mathbb{R}^n)$. For ordinary differential equation $\dot{x}(t) = \varphi(x(t))$ with the initial condition being $x(0) = x_0$. The transition for $\varphi \in \mathrm{BL}(E,\mathbb{R}^n)$ at time t can be defined as:

$$T_\varphi(t, x_0) = \{x(t) : \dot{x}(t) = \varphi(x(t)), x(0) = x_0\} \quad (3)$$

where $x(t)$ is the solution to $\dot{x}(t) = \varphi(x(t))$ with initial condition being $x(0) = x_0$. The transition start from a single point can also be applicable when $x(0)$ is a set: The transition at time t for a function $\varphi : E \mapsto \mathbb{R}^n$ with $E \subset \mathbb{R}^n$ starting from an initial set K_0 is:

$$T_\varphi(t, K_0) = \{x(t) : \dot{x}(t) = \varphi(x(t)), x(0) \in K_0\} \quad (4)$$

$T_\varphi(t, K_0)$ is a set rather than a single point, It is also a tube evolving from K_0 according to $\dot{x}(t) = \varphi(x(t))$.

2.3 Mutation equations for set evolutions

The transitions can be used to extend the time derivation of a function in the vector space to the time derivation of a tube in a general metric space. In the vector space, the time derivative for a function $f(t) : \mathbb{R}_{\geq 0} \mapsto \mathbb{R}^n$ is defined as

$$v = \lim_{\Delta t\to 0}\frac{f(t+\Delta t) - f(t)}{\Delta t} \quad (5)$$

This definition can be transformed to the first order approximation form, where the time derivative v for $f(t)$ should satisfy:

$$\lim_{\Delta t\to 0}\frac{1}{\Delta t}\|f(t+\Delta t) - (f(t) + v\Delta t)\| = 0 \quad (6)$$

where $f(t) + v\Delta t$ could be regarded as a new point that is start from $f(t)$ and then move along the direction of v after Δt time. In this case, we can conclude that if Eq. (6) is satisfied, then v can be seen as the derivative of $f(t)$ given in Eq. (5). Similarly, in the metric space $(\mathcal{K}(E), \mathcal{D})$, transition $T_\varphi(\Delta t, K(t))$ could be regarded as a new set in $\mathcal{K}(E)$ that is start from $K(t)$ and then move in the direction of $\varphi \in \mathrm{BL}(E,\mathbb{R}^n)$ after Δt time. Thus, similar to Eq. (6), φ satisfies the first order approximation of a tube $K(t)$ if:

$$\lim_{\Delta t\to 0}\frac{1}{\Delta t}\mathcal{D}(K(t+\Delta t), T_\varphi(\Delta t, K(t))) = 0 \quad (7)$$

where $K(t+\Delta t)$ is the set at time $t+\Delta t$ based on the tube $K(t) : \mathbb{R}_{\geq 0} \mapsto \mathcal{K}(\mathbb{R}^n)$. Based on the analogy from the vector space to the non-vector space, the derivative for a tube called mutation can be defined. Since there might be none, one or multiple φ satisfy Eq. (7) for a given tube, then the mutation is a set defined as a tube $K(t)$, denoted as $\overset{\circ}{K}(t)$, is defined as the collection of all $\varphi \in \mathrm{BL}(E,\mathbb{R}^n)$ satisfying Eq. (7):

$$\overset{\circ}{K}(t) = \{\varphi \in \mathrm{BL}(E,\mathbb{R}^n) : \text{Eq.(7) is satisfied}\} \quad (8)$$

From the above analysis, mutation equations in the non-vector space, the analogy to differential equations in the vector space, can be defined as follows:

For a given function $f : \mathcal{K}(E) \mapsto \mathrm{BL}(E,\mathbb{R}^n)$ mapping from a tube to a bounded Lipschitz function, the mutation equation for the tube is defined as:

$$f(K(t)) \in \overset{\circ}{K}(t) \text{ with } K(0) = K_0 \quad (9)$$

The solution to the mutation equation is the tube $K(t)$ such that the function $f(K(t)) \in \mathrm{BL}(E,\mathbb{R}^n)$ satisfies Eq. (7) at Lebesgue-almost every time.

3. LYAPUNOV FUNCTION FOR TUBES

3.1 Shape functionals and shape gradient

In order to design a visual tracking controller based on the non-vector space theory, we have to first introduce the concept of shape directional derivative for a shape functional over sets. For a subset E of $\mathbb{R}^n$ and a map $J : \mathcal{K}(E) \mapsto \mathbb{R}$, given a tube K in $\mathcal{K}(E)$ and mutation φ in $\mathrm{BL}(E,\mathbb{R}^n)$, the Eulerian derivative of the functional J at K in the directionφ is given by (see M. C. Delfour and J. -P. Zolsio (2011))

$$\overset{\circ}{J}(K)(\varphi) = \lim_{\Delta t\to 0^+}\frac{J(T_\varphi(\Delta t, K)) - J(K)}{\Delta t} \quad (10)$$

The limit given in Eq. (10) is called shape directional derivative in the given direction φ. To describe the difference between the initial set and the goal set for the visual target tracking problem, we will consider a special shape functional over two image sets

$$J(K,\hat{K}) = \int_K d^2_{\hat{K}}(x)\,dx = \int_K \|x - \Pi_{\hat{K}}(x)\|^2 dx \quad (11)$$

where K and $\hat{K}$ are the two tubes representing the initial image set and the goal image set, $\Pi_{\hat{K}}(x)$ is the projection on $\hat{K}$ of x as given in Eq. (2). Let $T_\varphi(\Delta t, K)$ be the transition starting from set K along direction φ and $T_{\hat{\varphi}}(\Delta t, \hat{K})$ the transition in starting

from set $\hat{K}$ along direction $\hat{\varphi}$ respectively, then the 2-sets shape functional after $\triangle t$ time could be given as

$$J(T_{\varphi}(\Delta t,K),T_{\hat{\varphi}}(\Delta t,\hat{K}))=\int_{T_{\varphi}(\Delta t,K)}d^2_{T_{\hat{\varphi}}(\Delta t,\hat{K})}(x)dx \tag{12}$$

By transforming the integral in Eq. (12) to an integral over tube K leads to

$$\begin{aligned}&\int_{T_{\varphi}(\Delta t,K)}d^2_{T_{\hat{\varphi}}(\Delta t,\hat{K})}(x)dx\\&=\int_K\|T_{\varphi}(\Delta t,x)-\Pi_{T_{\hat{\varphi}}(\Delta t,\hat{K})}\left(T_{\varphi}(\Delta t,x)\right)\|^2dx\end{aligned} \tag{13}$$

then based on Eq. (10) the shape directional derivative of the shape function in Eq. (11) along two directions φ and $\hat{\varphi}$ could be given as

$$\begin{aligned}&\overset{\circ}{J}(K,\hat{K})(\varphi,\hat{\varphi})\\&=\lim_{\Delta t\to 0}\frac{J(T_{\varphi}(\Delta t,K),T_{\hat{\varphi}}(\Delta t,\hat{K}))-J(K,\hat{K})}{\Delta t}=\\&\lim_{\Delta t\to 0}\frac{\int_K(\|T_{\varphi}(\Delta t,x)-\Pi_{T_{\hat{\varphi}}(\Delta t,\hat{K})}(T_{\varphi}(\Delta t,x))\|^2-\|x-\Pi_{\hat{K}}(x)\|^2)dx}{\Delta t}\\&=\int_K\lim_{\Delta t\to 0}\frac{\|T_{\varphi}(\Delta t,x)-\Pi_{T_{\hat{\varphi}}(\Delta t,\hat{K})}(T_{\varphi}(\Delta t,x))\|^2-\|x-\Pi_{\hat{K}}(x)\|^2}{\Delta t}dx\end{aligned} \tag{14}$$

Knowing that $x=T_{\varphi}(0,x)$, $\Pi_{\hat{K}}(x)=\Pi_{T_{\hat{\varphi}}(0,\hat{K})}(T_{\varphi}(0,x))$, then Eq. (14) can be expressed as

$$\begin{aligned}&\overset{\circ}{J}(K,\hat{K})(\varphi,\hat{\varphi})=\\&\int_K(\|T_{\varphi}(\Delta t,x)-\Pi_{T_{\hat{\varphi}}(\Delta t,\hat{K})}(T_{\varphi}(\Delta t,x))\|^2)'_{\Delta t=0}dx\end{aligned} \tag{15}$$

The expression in Eq. (15) is the directional derivative of the shape functional $J(K,\hat{K})$ over two tubes K and $\hat{K}$ along the two directions φ and $\hat{\varphi}$. This is different from the directional derivative given in Eq. (10) that is a shape functional over single tube K and along a single direction φ, which could be given as

$$\begin{aligned}&\overset{\circ}{J}(K)(\varphi)=\\&\int_K(\|T_{\varphi}(\Delta t,x)-\Pi_{\hat{K}}(T_{\varphi}(\Delta t,x))\|^2)'_{\Delta t=0}dx\end{aligned} \tag{16}$$

Based on the definition of the transition in Eq. (3), we can see that $T_{\varphi}(\Delta t,x)$ could be seen as a mapping from φ to the point $x(\Delta t)$. Let $d^2_{\hat{K}}(x)$ be denoted as $f(x,\hat{K})$, then based on product rule, chain rule and Gau theorem (M. C. Delfour and J. -P. Zolsio (2011)), we have

$$\begin{aligned}&\overset{\circ}{J}(K)(\varphi)=\int_K(f(x,\hat{K})\circ\varphi(x))'dx\\&=\int_K(\nabla f(x,\hat{K})^T\varphi(x)+f(x,\hat{K})\mathrm{div}\varphi(x))dx\end{aligned} \tag{17}$$

where " $\circ$ " represents the composite mapping, divφ is the divergence of the vector φ. For the visual tracking problem in this paper, the mutation under the two transitions $T_{\varphi}(\Delta t,x)$ and $T_{\hat{\varphi}}(\Delta t,x)$ can be regarded as two separate transitions by first moving initial image set K_0 under the mutation φ with the target set being static, then move the target image set under $\hat{\varphi}$ with the moved initial image set $T_{\varphi}(\Delta t,K_0)$ being static, then for the directional derivative in Eq. (15), we have

$$\overset{\circ}{J}(K,\hat{K})(\varphi,\hat{\varphi})=\overset{\circ}{J}(K,\hat{K})(\varphi)+\overset{\circ}{J}(K,\hat{K})(\hat{\varphi}) \tag{18}$$

Proof: For a mapping $f(x,y):\mathcal{K}(E)\times\mathcal{K}(E)\mapsto\mathbb{R}$, with $x,y\in\mathbb{R}^n, E\subset\mathbb{R}^n$, given a mutation $\Phi=(\varphi,\hat{\varphi})$ in $\mathrm{BL}(E,\mathbb{R}^{2n})$, let $z=(x,y)$, $x=(x_1,x_2,\cdots,x_n)$, $y=(y_1,y_2,\cdots,y_n)$, $\varphi=(u_1,u_2,\cdots,u_n)$, $\hat{\varphi}=(v_1,v_2,\cdots,v_n)$. Based on Eq.(17), the shape directional derivative of f along "direction" Φ can be given as

$$\begin{aligned}&\overset{\circ}{f}(z)(\Phi)=\nabla f(z)^T\Phi+f(z)\mathrm{div}\Phi\\&=(\frac{\partial f}{\partial x_1}u_1+\frac{\partial f}{\partial x_2}u_2+\cdots+\frac{\partial f}{\partial x_n}u_n+\frac{\partial f}{\partial y_1}v_1+\\&\frac{\partial f}{\partial y_2}v_2+\cdots+\frac{\partial f}{\partial y_n}v_n)+f(z)(\frac{\partial u_1}{\partial x_1}+\frac{\partial u_2}{\partial x_2}+\cdots\\&+\frac{\partial u_n}{\partial x_n}+\frac{\partial v_1}{\partial y_1}+\frac{\partial v_2}{\partial y_2}+\cdots+\frac{\partial v_n}{\partial y_n})\end{aligned} \tag{19}$$

Eq. (19) can also be written as

$$\begin{aligned}&\overset{\circ}{f}(z)(\Phi)=(\frac{\partial f}{\partial x_1}u_1+\frac{\partial f}{\partial x_2}u_2+\cdots+\frac{\partial f}{\partial x_n}u_n)+\\&(\frac{\partial f}{\partial y_1}v_1+\frac{\partial f}{\partial y_2}v_2+\cdots+\frac{\partial f}{\partial y_n}v_n)+\\&f(z)(\frac{\partial u_1}{\partial x_1}+\frac{\partial u_2}{\partial x_2}+\cdots+\frac{\partial u_n}{\partial x_n})+\\&f(z)(\frac{\partial v_1}{\partial y_1}+\frac{\partial v_2}{\partial y_2}+\cdots+\frac{\partial v_n}{\partial y_n})\\&=\overset{\circ}{f}(z)(\varphi)+\overset{\circ}{f}(z)(\hat{\varphi})\end{aligned} \tag{20}$$

Then Eq. (18) holds. Similarly, $T_{\hat{\varphi}}(\Delta t,x)$ is a mapping from $\hat{\varphi}$ to the point $x(\Delta t)$. Then Eq. (18) can be further expressed as

$$\begin{aligned}&\overset{\circ}{J}(K,\hat{K})(\varphi,\hat{\varphi})\\&=\int_K(f(T_{\varphi}(0,x),T_{\hat{\varphi}}(0,\hat{K}))\circ\varphi(x))'dx\\&+\int_K(f(T_{\varphi}(0,x),T_{\hat{\varphi}}(0,\hat{K}))\circ\hat{\varphi}(x))'dx\\&=\int_K(\nabla f(x,\hat{K})^T\varphi(x)+f(x,\hat{K})\mathrm{div}\varphi(x))dx\\&+\int_K(\dot{f}(x,\hat{K})(\hat{\varphi}(x))+f(x,\hat{K})\mathrm{div}\hat{\varphi}(x))dx\end{aligned} \tag{21}$$

where " $\circ$ " represents the composite mapping. $\mathrm{div}\Psi(x)=\sum_{i=1}^n\partial\Psi_i(x)/\partial x_i$, $\Psi_i(x), i=1,2,\cdots,m$ are the m row vectors in $\Psi\in\mathbb{R}^{m\times n}$ and $\partial\Psi_i(x)/\partial x_i$ is also a m-dimension row vector, Ψ represents φ or $\hat{\varphi}$. divΨ is the divergence of the vector Ψ.

In Eq. (21), the gradient $\nabla f(x,\hat{K})=\nabla d^2_{\hat{K}}(x)$ represents maximum decrease direction for distance of point $x\in K$ to the set $\hat{K}$, when the set K is dynamic and set $\hat{K}$ is static, this gradient can be given as (see (Luc Doyen (1995)) for proof)

$$\nabla d^2_{\hat{K}}(x)=2(x-\Pi_{\hat{K}}(x)), x\in K \tag{22}$$

The material derivative (M. C. Delfour and J. -P. Zolsio (2011)) $\dot{f}(x,\hat{K})(\hat{\varphi}(\hat{K}))$ could be given as

$$\dot{f}(x,\hat{K})(\hat{\varphi}(\hat{K}))=\lim_{\Delta t\to 0}\frac{f(x,T_{\hat{\varphi}}(\Delta t,\hat{K}))-f(x,\hat{K})}{\Delta t} \tag{23}$$

For the calculation of the material derivative, there may exist three special cases:

Case 1: The target is a fixed image as shown in Fig. 1 . In this case, the tracking problem becomes an image-aliment problem. The fixed image could be regarded as an equilibrium point for this control problem and the controller can be designed based on Doyen's approach (Luc Doyen (1995)). In this case, each element x in the fixed image $\hat{K}$ satisfies $\hat{\varphi}(x)=0$.

Case 2: The target set $\hat{K}$ is moving as a rigid body along a prescribed trajectory, as shown in Fig. 2. In this case, the mutation equation $\hat{\varphi}(\hat{K})$ could be expressed with the form

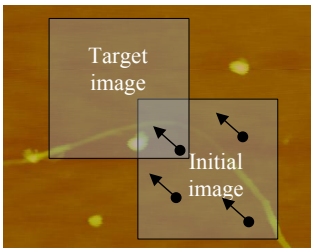

Fig. 1. The fixed target image and the initial image

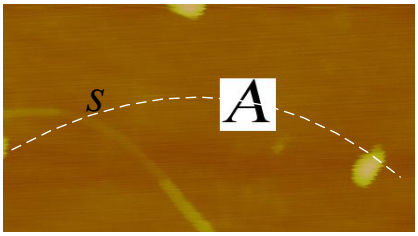

Fig. 2. A rigid body target moving along a trajectory

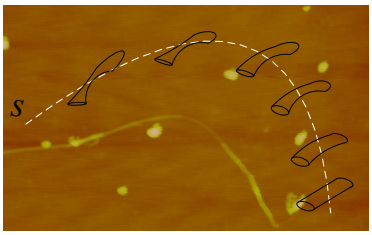

Fig. 3. The target moving along a trajectory as a flexible body

$\hat{\varphi}(\hat{K}) = L(\hat{K})v$, with v being the velocity vector of the target $\hat{K}$ as a rigid body. Then

$$\dot{f}(x,\hat{K})(\hat{\varphi}(\hat{K})) = \nabla f(x,\hat{K})^T L(\hat{K})v \tag{24}$$

In this paper, we will focus on this case, i.e., the target moves as a rigid body.

Case 3: The target set $\hat{K}$ is moving as a non-rigid body along a trajectory, as shown in Fig. 3. In this case, the mutation equation $\hat{\varphi}(\hat{K})$ could not be expressed with the form $\hat{\varphi}(\hat{K}) = L(\hat{K})v$, i.e., it is no longer a linear system with respective to the velocity of target image set. Then we could not design such a controller based on the α-Lyapunov function as given in section 3.2.

3.2 Lyapunov functions for tubes

In order to steer $K(t)$ to follow a given shape $\hat{K}(t)$, Let us use the shape Lyapunov function as follow

$$V(K,\hat{K}) = \int_K d^2_{\hat{K}}(x)dx + \int_{\hat{K}} d^2_K(x)dx \tag{25}$$

It is obvious that $V(K,\hat{K}) = 0$ implies that $K = \hat{K}$. It could be treated as the Hausdorff distance between the two sets K and $\hat{K}$. By differentiating the Lyapunov function in the direction φ and $\hat{\varphi}$ we have

$$\begin{aligned}
&\mathring{V}(K,\hat{K})(\varphi,\hat{\varphi})\\
&= \int_K \nabla d^2_{\hat{K}}(x)^T\varphi(x)dx + \int_K d^2_{\hat{K}}(x)\mathrm{div}\varphi(x)dx\\
&+ \int_K \nabla d^2_{\hat{K}}(x)^T\hat{\varphi}(x)dx + \int_K d^2_{\hat{K}}(x)\mathrm{div}\hat{\varphi}(x)dx\\
&+ \int_{\hat{K}} \nabla d^2_K(x)^T(-\varphi(\Pi_K(x)))dx + \int_{\hat{K}} d^2_K(x)\mathrm{div}\varphi(x)dx\\
&+ \int_{\hat{K}} \nabla d^2_K(x)^T(-\hat{\varphi}(\Pi_K(x)))dx + \int_{\hat{K}} d^2_K(x)\mathrm{div}\hat{\varphi}(x)dx
\end{aligned} \tag{26}$$

we also have

$$\begin{aligned}
&\mathring{V}(K,\hat{K})(\varphi,\hat{\varphi})\\
&= \int_K \left\langle x-\Pi_{\hat{K}}(x), 2\varphi(x) + (x-\Pi_{\hat{K}}(x))\mathrm{div}\varphi(x)\right\rangle dx\\
&+ \int_K \left\langle x-\Pi_{\hat{K}}(x), 2\hat{\varphi}(x) + (x-\Pi_{\hat{K}}(x))\mathrm{div}\hat{\varphi}(x)\right\rangle dx\\
&+ \int_{\hat{K}} \left\langle x-\Pi_K(x), -2\varphi(\Pi_K(x)) + (x-\Pi_K(x))\mathrm{div}\varphi(x)\right\rangle dx\\
&+ \int_{\hat{K}} \left\langle x-\Pi_K(x), -2\hat{\varphi}(\Pi_K(x)) + (x-\Pi_K(x))\mathrm{div}\hat{\varphi}(x)\right\rangle dx
\end{aligned} \tag{27}$$

When $\varphi(x) = L(x)u$ is the mapping from camera velocity to the pixel velocity, $\hat{\varphi}(x) = -L(x)v$ is the mapping from target's velocity to the pixel velocity. If the velocity of the moving target is measurable, i.e., v is known, then based on Doyen's approach (Luc Doyen (1995)), we can construct the α-Lyapunov function as

$$\begin{aligned}
&\mathring{V}(K,\hat{K})(\varphi,\hat{\varphi}) =\\
&\int_K \left\langle x-\Pi_{\hat{K}}(x), 2L(x)u + (x-\Pi_{\hat{K}}(x))\mathrm{div}L(x)u\right\rangle dx\\
&+ \int_K \left\langle x-\Pi_{\hat{K}}(x), -2L(x)v - \left(x-\Pi_{\hat{K}}(x)\right)\mathrm{div}L(x)v\right\rangle dx\\
&+ \int_{\hat{K}} \left\langle x-\Pi_K(x), -2L(\Pi_K(x))u + (x-\Pi_K(x))\mathrm{div}L(x)u\right\rangle dx\\
&+ \int_{\hat{K}} \left\langle x-\Pi_K(x), 2L(\Pi_K(x))v - (x-\Pi_K(x))\mathrm{div}L(x)v\right\rangle dx
\end{aligned} \tag{28}$$

then we have

$$\begin{aligned}
&\int_K \left\langle x-\Pi_{\hat{K}}(x), D_1(x)(u-v)\right\rangle dx\\
&-\int_{\hat{K}} \left\langle x-\Pi_K(x), D_2(\Pi_K(x))(u-v)\right\rangle dx \le\\
&-\alpha\int_K \left\|x-\Pi_{\hat{K}}(x)\right\|^2 dx - \alpha\int_{\hat{K}} \left\|x-\Pi_K(x)\right\|^2 dx
\end{aligned} \tag{29}$$

where $D_1(x) = 2L(x) + (x-\Pi_{\hat{K}}(x))divL(x), D_2(x) = -2L(\Pi_K(x)+(x-\Pi_K(x))\mathrm{div}L(x)$. Eq. (29) could be written in the following form

$$\begin{aligned}
&\langle \int_K D_1^*(x)(x-\Pi_{\hat{K}}(x))dx+\\
&\int_{\hat{K}} D_2^*(x)(x-\Pi_K(x))dx, u-v\rangle \le -\alpha V(K,\hat{K})
\end{aligned} \tag{30}$$

where D_1^* and D_2^* are the transpose of D_1 and D_2 respectively. Based on Doyen's approach (Luc Doyen (1995)), a solution of this linear problem is given by

$$u(K,\hat{K}) = -\alpha A(K,\hat{K})^+ V(K,\hat{K}) + v \tag{31}$$

where

$$A(K,\hat{K})^+ = \frac{A^T(K,\hat{K})}{A^T(K,\hat{K})A(K,\hat{K})} \tag{32}$$

and

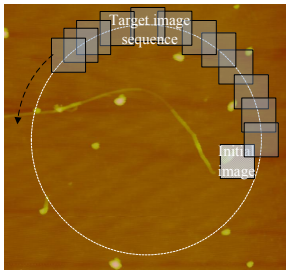

Fig. 4. The sequence of the target images along a circle and the initial position of the initial image

$$\begin{aligned} A(K,\hat{K}) &= \int_K d^2_{\hat{K}}(x)(\sum_{i=1}^{m}\partial L_i(x)/\partial x_i)dx \\ &+2\int_K (x-\Pi_{\hat{K}}(x))^T L(x)dx \\ &-2\int_{\hat{K}} (x-\Pi_K(x))^T L(\Pi_K(x))dx \\ &+\int_{\hat{K}} d^2_K(x)(\sum_{i=1}^{m}\partial L_i(x)/\partial x_i)dx \end{aligned} \tag{33}$$

4. APPLICATION OF THE PROPOSED APPROACH

In order to control the robot based on the vision information, we first capture every video frame from the camera, then the image frames are transferred into gray scale images. Suppose that each pixel in the image plane can be expressed as

$$x = [x_1, x_2, x_3] \tag{34}$$

In constant lighting condition, for each pixel we have $\dot{x}_3 = 0$. Each point x in the image plane corresponding to a 3D point p in reality, and the 3D point in reality expressed in camera frame is given as

$$p = [p_x, p_y, p_z] \tag{35}$$

Using the projective geometry, we have

$$x_1 = \lambda\frac{p_x}{p_z}, x_2 = \lambda\frac{p_y}{p_z} \tag{36}$$

Where λ is the focal length. Let the velocity of the camera be denoted as $u(t) = [v_x, v_y, v_z, \omega_x, \omega_y, \omega_z]^T$, then we have the following relation (Chaumette and Hutchinson (2006))

$$\dot{x} = [\dot{x}_1, \dot{x}_2, \dot{x}_3]^T = L(x)\,u(t) \tag{37}$$

where

$$L = \begin{bmatrix} -\frac{1}{p_z} & 0 & \frac{x_1}{p_z} & x_1x_2 & -(1+x_1^2) & x_2 \\ 0 & -\frac{1}{p_z} & \frac{x_2}{p_z} & 1+x_2^2 & -x_1x_2 & -x_1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{38}$$

From the expression of L, we have:

$$\sum_{i=1}^{3}\frac{\partial L_i}{\partial x_i} = [\,0\;\;0\;\;2/p_z\;\;3x_2\;\;-3x_1\;\;0\,] \tag{39}$$

In this paper, we will conduct the experiments for the 2D translation. Let K be the set of the image captured from the camera and $\hat{K}$ be the set of the goal image. As shown in Fig. 4, suppose that the moving target is the square area along a circle with 80×80 pixels and moving with constant angular velocity around a fixed point. Then our object is to steer the current image to follow the sequence of the target images.

In order to give an intuitive insight for the algorithm, we first show that the visual tracking approach proposed in this paper is

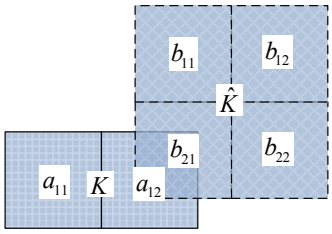

Fig. 5. Two sets of different size with one element as overlap

applicable for the two sets with different size, an example is as shown in Fig. 5, the set K has two elements, $a_{11} = 100, a_{12} = 110$, the set $\hat{K}$ has four elements, $b_{11} = 105, b_{12} = 110, b_{21} = 115, b_{22} = 120$, i.e., element a_{12} in set K is identical of the element b_{12} in set $\hat{K}$, i.e., the overlap elements of the two sets are a_{12} and b_{21}. Then the distances between each element in set K to each element in set $\hat{K}$ could be calculated as follows

$$\begin{array}{l|l} \|a_{11}b_{11}\|^2 = 25 & \|b_{11}a_{11}\|^2 = 25 \\ \|a_{11}b_{12}\|^2 = 101 & \|b_{11}a_{12}\|^2 = 26 \\ \|a_{11}b_{21}\|^2 = 226 & \|b_{12}a_{11}\|^2 = 101 \\ \|a_{11}b_{22}\|^2 = 402 & \|b_{12}a_{12}\|^2 = 0 \\ \|a_{12}b_{11}\|^2 = 26 & \|b_{21}a_{11}\|^2 = 226 \\ \|a_{12}b_{12}\|^2 = 0 & \|b_{21}a_{12}\|^2 = 27 \\ \|a_{12}b_{21}\|^2 = 27 & \|b_{22}a_{11}\|^2 = 402 \\ \|a_{12}b_{22}\|^2 = 101 & \|b_{22}a_{12}\|^2 = 101 \end{array} \tag{40}$$

From the distances given in Eq. (40), we can see that the projections of elements a_{11} and a_{12} in set $\hat{K}$ are b_{11} and b_{12} respectively, and the projections of the elements b_{11}, b_{12}, b_{21} and b_{22} in set K are a_{11}, a_{12}, a_{12} and a_{12} respectively. If K is moving based on mutation φ with $\hat{K}$ being static, then the directional derivative can be calculated as

$$\begin{aligned} \mathring{J}(K,\hat{K})(\varphi) &= \int_K (\nabla f(x)^T\varphi(x) + f(x)\mathrm{div}(\varphi(x)))dx \\ &= 2(a_{11}-b_{11})^T\varphi(a_{11}) + 2(a_{12}-b_{12})^T\varphi(a_{12}) \\ &+ \|a_{11}b_{11}\|^2\,\mathrm{div}\varphi(a_{11}) + \|a_{12}b_{11}\|^2\,\mathrm{div}\varphi(a_{12}) \end{aligned} \tag{41}$$

From Eq. (41) and the controller terms given in Eq. (31) ∼ Eq. (33), we could conclude that two sets without identical overlaps can also lead to the convergence of the controller.

Based on the above example, we divide the view area of camera into four sub-area as shown in Fig. 6, then we put a target image in the centre of the view area (view centre), which is regarded as an initial image set, and then the four sub-areas 1, 2, 3 and 4 of the camera view will be treated as for target images (the current image has different size with the initial image in view center), a 4-step searching are conducted to see the target locate at which sub-area, after that, if the target is located at sub-area 1, for example, then this sub-area is used as the target image set, which is also used together with the centre view of the camera to calculated the control velocity at this situation. The searching only need to calculated for once if we don't know where is the target, but this treatment could improve the robustness of the control algorithm.

Special attention has to be paid to the above treatment, the target image is put into the center of the whole image that captured from the camera, then intensive value in the same position may be different, they only share the same coordinate index.

We first apply the controller of Doyen (Luc Doyen (1995)) to the trajectory tracking problem as given in Fig. 4, then we

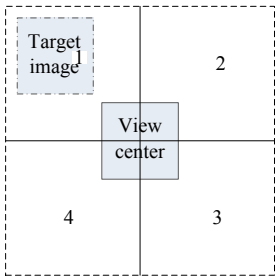

Fig. 6. Tracking a target in the view area of the camera

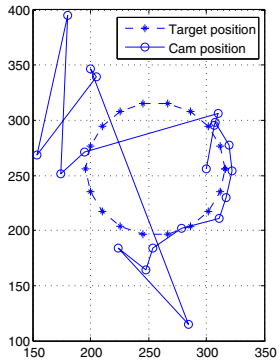

Fig. 7. The camera image fail to follow the target during tracking process

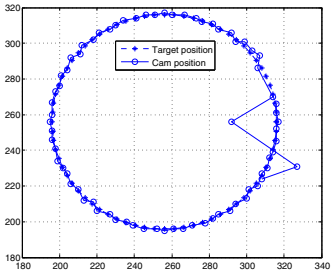

Fig. 8. The tracking trajectory with target moving in 5 degree/iteration

plot the trajectory of the target images and the sequence of the position of the initial image, as shown in Fig. 7 , in this case, the controller is invalid in this case because the calculated control input for the camera image is invalid, because the target moving so quickly such that the camera image always doesn't have intersection with the target image. Therefore, this controller is not suitable for the tracking problem.

By using the controller as given in Eq. (31), The moving trajectory of the target image and the corresponding tracking trajectory of the camera captured image are as shown in Fig. 8 . In the tracking process, the Lyapunov function value and the Hausdorff distance is as shown in Fig. 9 .

5. CONCLUSIONS

This paper has proposed a featureless visual tracking approach based on the non-vector space theory. By considering the image as a set, we have derived the set dynamic model using the mutation equations. Different from existing non-vector space control approach, both the target image and the initial image in visual tracking problem are dynamic set during the control process, a shape functional for describing the difference of

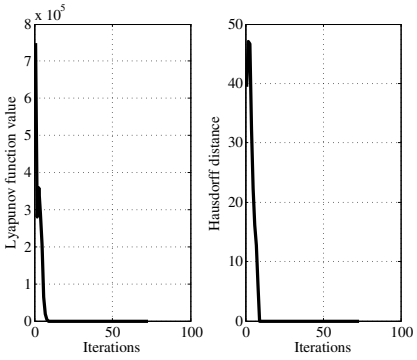

Fig. 9. The Lyapunov function value and Hausdorff distance during tracking

the two dynamic sets is created, the directional derivative of this shape functional has been derived and this derivative can be considered as the error dynamic of the tracking problem. Based on the stabilized Lyapunov function, we can finally design a controller for the online visual tracking. This image-based visual tracking controller does not require any feature extraction during the control process. Experiments on tracking the target moving in 2D translation has been conducted to illustrate the feasibility of the problem.

REFERENCES

Amaury Dame and Eric Marchand. (2011). Mutual information-based visual servoing. *IEEE Transactions on Robotics*, 27(5), pp.1-12.

Chaumette and Hutchinson. (2006). Visual Servo Control, Part I: Basic Approaches. *IEEE Transactions on Robotics and Automation*, 13(4), pp. 82-90.

Christophe Collewet and Eric Marchand. (2011). Photometric visual servoing. *IEEE Transactions on Robotics*, 27(4), pp.828-834.

E. Marchand and F. Chaumette. (2005). Feature tracking for visual servoing purposes. *Robotics and Autonomous Systems*, 52(1), pp.53-70.

F. Chaumette and S. Hutchinson. (2007). Visual servo control part II: Advanced approaches. *IEEE Robotics and Automation Magazine*, 14(1), pp.109-118.

Jean Pierre Aubin. (1998). *Mutational and Morphological Analysis: Tools for Shape Evolution and Morphogenesis*. Birkhäuser.

J. Zhao, Yunyi Jia, Ning Xi, Weixian Li, Bo Song, and Liang Sun. (2012). Visual servoing using non-vector space control theory. *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pp.1-6.

K. Deguchi. (2000). A direct interpretation of dynamic images with camera and object motions for vision guided robot control. *Int. Journal of Computer Vision*, 37(1), pp. 7-20.

Luc Doyen. (1995). Mutational equations for shapes and vision-based control. *Journal of Mathematical Imaging and Vision*, 5(2), pp.99-109.

M. C. Delfour and J. -P. Zolsio. (2011). *Shapes and Geometries-Metrics, Analysis, Differential Calculus, and Optimization*. 2nd, SIAM, Philadelphia.

Vinutha Kallem, Maneesh Dewan, John P. Swensen, Gregory D. Hager, and Noah J. Cowan. (2007). Kernel-based visual servoing. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp.1975-1980.