A Sliding Mode Approach to Visual Motion Estimation

Eray Dogan Burak Yilmaz Mustafa Unel Asif Sabanovic

Mechatronics Program, Faculty of Engineering and Natural Sciences,
Sabanci University, Orhanli-Tuzla, Istanbul, 34956, TURKEY
{*eraydogan,burakyilmaz*}*@su.sabanciuniv.edu* {*munel,asif*}*@sabanciuniv.edu*

***Abstract*— The problem of estimating motion from a sequence of images has been a major research theme in machine vision for many years and remains one of the most challenging ones. In this work, we use sliding mode observers to estimate the motion of a moving body with the aid of a CCD camera. We consider a variety of dynamical systems which arise in machine vision applications and develop a novel identification procedure for the estimation of both constant and time varying parameters. The basic procedure introduced for parameter estimation is to recast image feature dynamics linearly in terms of unknown parameters and construct a sliding mode observer to produce asymptotically correct estimates of the observed image features, and then use "equivalent control" to explicitly compute parameters. Much of our analysis has been substantiated by computer simulations and real experiments.**

I. Introduction

Motion parameter estimation in machine vision has been initiated by the early works of Ullman [11]. Roach and Aggarwal [18] tested the estimation problem with real images. Nagel [19] reduced the problem to the solution of a nonlinear equation. An analytical solution for eight feature points was given independently by Longuet-Higgins [20], and Tsai and Huang [21]. Zhuang [22], [23] presented a simplified eight point algorithm and considered the uniqueness issue. In the uniqueness context, Netravali et.al., [24] introduced the homotopy method and showed the existence of ten solutions. Faugeras and Maybank [9] showed that at most ten solutions can be obtained from five feature points using projective geometry. Using the quaternions, Jerian and Jain [25] reduced the problem to solving the resultant of a pair of quartic binary polynomials. They also compared known algorithms based on their performances with noisy data [26].

Many of the existing algorithms in the literature perform poorly under noisy data. A robust algorithm was introduced by Weng et.al., [27] and by Spetsakis and Aloimonos [16], [17]. They used optimization based methods to compute epipolar equations. Grzywacz and Hildreth [28] have also shown that the effects of image noise on reconstruction from image velocities are severe in some cases. Jerian and Jain [25], Murray and Buxton [29] proposed various schemes towards a stable reconstruction algorithm. The particular estimation problem has been summarized in two books by Maybank [30] and by Kanatani [13]. For some other references we would like to refer to [37], [40], [38]. Ghosh et.al., [35], [36] has shown

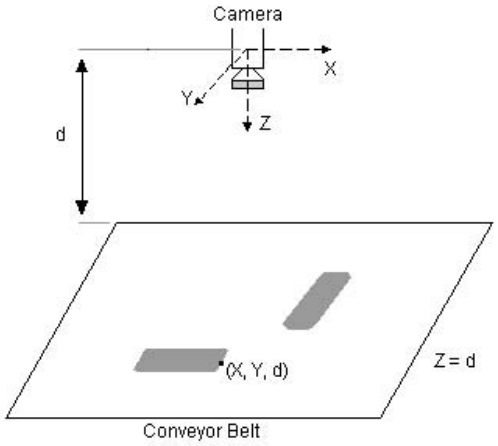


Fig. 1. Rigid motion of an object on a conveyor belt under a scaled orthographic camera

that the problem of motion and shape estimation for a moving textured surface can always be analyzed as a specific parameter estimation problem of a perspective system. A perspective system is a linear system with a homogeneous observation function (see [34]). The specific form of the perspective system depends on how the surface and the motion field have been parameterized.

In this paper, we consider a variety of dynamical systems which arise in machine vision applications and develop a novel identification procedure for the estimation of both constant and time varying parameters. We will be dealing with the 'feature based analysis' (see [32], [31], [39], [41], [14]). The basic procedure introduced for parameter estimation is to recast image feature dynamics linearly in terms of unknown parameters and construct a sliding mode observer [42], [43] to produce asymptotically correct estimates of the observed image features, and then use "equivalent control" to explicitly compute parameters.

II. Camera, Motion and Structure Models

In this section, we describe models for the camera imaging geometry, the motion of the scene relative to the camera, and the structure of rigid surfaces in the scene. We assume that a stationary camera is viewing a rigid surface which is moving

smoothly in 3D. The same analysis applies to the case of a moving camera viewing a stationary surface.

A. Camera Projection Model

Let 3D points on a rigid surface be defined with respect to the camera reference frame, with the depth axis is aligned with the optical axis and perpendicular to the image plane. The image plane is at a distance f, the focal length, from the center of projection (COP). The perspective projection of a 3D surface point (X, Y, Z) onto an image point (x, y) is the given by

$$x = f\frac{X}{Z}, \qquad y = f\frac{Y}{Z}. \tag{1}$$

Note that when the focal length becomes quite large, i.e. $f \to \infty$, the perspective projection is reduced to an orthographic projection given by

$$x = X, \qquad y = Y \tag{2}$$

When the depth of objects, δz, is much smaller than their average distance, $\bar{Z}$, from the camera along the optical axis, i.e. $|\delta z| \ll \bar{Z}$, the full perspective camera model can be replaced by the weak-perspective or so called scaled orthographic camera model. An example for this is the camera located at a sufficiently large distance, d, above a plane of a conveyor belt, with moving piece-parts over it as in Figure 1. In this case $\bar{Z} = d$. The perspective projection given in (6) takes the following form:

$$x = f\frac{X}{\bar{Z}} = f\frac{X}{d}, \qquad y = f\frac{Y}{\bar{Z}} = f\frac{Y}{d} \tag{3}$$

B. Motion Models

If a rigid body is moving with instantaneous translational velocity, T, and rotational velocity, Ω, then the 3D instantaneous velocity of points on the surface is given by

$$\begin{pmatrix}\dot{X}\\ \dot{Y}\\ \dot{Z}\end{pmatrix} = \Omega \times \begin{pmatrix}X\\ Y\\ Z\end{pmatrix} + T \tag{4}$$

$$\begin{pmatrix}\dot{X}\\ \dot{Y}\\ \dot{Z}\end{pmatrix} = \underbrace{\begin{pmatrix}0 & -\omega_3 & \omega_2\\ \omega_3 & 0 & -\omega_1\\ -\omega_2 & \omega_1 & 0\end{pmatrix}}_{\triangleq\, [\Omega]_\times} \begin{pmatrix}X\\ Y\\ Z\end{pmatrix} + \underbrace{\begin{pmatrix}b_1\\ b_2\\ b_3\end{pmatrix}}_{\triangleq\, T} \tag{5}$$

where $\Omega = (\omega_1, \omega_2, \omega_3)^T$ and $T = (b_1, b_2, b_3)^T$. Note that $[\Omega]_\times$ is a skew-symmetric, i.e. $[\Omega]_\times + [\Omega]_\times^T = 0$, obtained from Ω.

The motion of the projected points in the image plane, $(\dot{x}, \dot{y})$, can be obtained from:

$$x = f\frac{X}{Z}, \qquad y = f\frac{Y}{Z}. \tag{6}$$

$$\begin{pmatrix}\dot{x}\\ \dot{y}\end{pmatrix} = f\begin{pmatrix}\frac{\dot{X}-x\dot{Z}}{Z}\\ \frac{\dot{Y}-y\dot{Z}}{Z}\end{pmatrix}, \tag{7}$$

and using (4) gives the required 2D motion equations:

$$\dot{x} = f\omega_2 x^2 - f\omega_1 xy - f\omega_3 y + f\omega_2 + f(b_1 - xb_3)/Z, \tag{8}$$

$$\dot{y} = -f\omega_1 y^2 + f\omega_2 xy + f\omega_3 x - f\omega_1 + f(b_2 - yb_3)/Z. \tag{9}$$

C. Planar Structure

Note the existence of depth variable, Z, in the motion field described in (8)-(9). Depth affects the image dynamics. If we assume a smooth surface structure, locally a linear depth variation can be used. This means that locally about a point (X, Y, Z), the surface is approximately planar, i.e. $Z \approx pX + qY + r$, with the orientation defined by the surface normal, $N = (p, q)^T$.

Now the depth variable Z can be eliminated from motion fields by plugging

$$Z \approx pX + qY + r \Rightarrow 1 \approx \frac{pX + qY + r}{Z} = px + qy + r/Z$$
$$\Rightarrow 1/Z \approx \frac{1 - (px + qy)}{r}$$

into (8)-(9). This substitution yields so called Riccati dynamics:

$$\frac{d}{dt}\begin{pmatrix}x\\ y\end{pmatrix} \approx \begin{pmatrix}d_1\\ d_2\end{pmatrix} + \begin{pmatrix}d_3 & d_4\\ d_5 & d_6\end{pmatrix}\begin{pmatrix}x\\ y\end{pmatrix} + \begin{pmatrix}d_7x^2 + d_8xy\\ d_8y^2 + d_7xy\end{pmatrix} \tag{10}$$

Image dynamics are governed by differential equations with quadratic right hand sides. Previous work on motion estimation suggests that, in practice, the second order terms play a minor role, to the extent that they often end up fitting the noise or deviations from the planar model [1],[33]. Therefore an affine image dynamics of the form

$$\frac{d}{dt}\begin{pmatrix}x\\ y\end{pmatrix} = \begin{pmatrix}a_1 & a_2\\ a_3 & a_4\end{pmatrix}\begin{pmatrix}x\\ y\end{pmatrix} + \begin{pmatrix}b_1\\ b_2\end{pmatrix} \tag{11}$$

is usually considered.

Note also that under the scaled-orthographic projection described in (3), a 2D rigid motion

$$\begin{pmatrix}\dot{X}\\ \dot{Y}\end{pmatrix} = \begin{pmatrix}0 & -\omega\\ \omega & 0\end{pmatrix}\begin{pmatrix}X\\ Y\end{pmatrix} + \begin{pmatrix}b_1\\ b_2\end{pmatrix} \tag{12}$$

will imply the following motion on the image plane:

$$\begin{pmatrix}\dot{x}\\ \dot{y}\end{pmatrix} = \frac{f}{d}\begin{pmatrix}\dot{X}\\ \dot{Y}\end{pmatrix} = \frac{f}{d}\left[\begin{pmatrix}0 & -\omega\\ \omega & 0\end{pmatrix}\begin{pmatrix}X\\ Y\end{pmatrix} + \begin{pmatrix}b_1\\ b_2\end{pmatrix}\right]$$
$$= \begin{pmatrix}0 & -\omega\\ \omega & 0\end{pmatrix}\begin{pmatrix}fX/d\\ fY/d\end{pmatrix} + \frac{f}{d}\begin{pmatrix}b_1\\ b_2\end{pmatrix} = \begin{pmatrix}0 & -\omega\\ \omega & 0\end{pmatrix}\begin{pmatrix}x\\ y\end{pmatrix} + \begin{pmatrix}\bar{b}_1\\ \bar{b}_2\end{pmatrix} \tag{13}$$

where, clearly the only difference is in the translational parameters, thus making the image dynamics also of rigid motion type.

III. Composite Image Feature Dynamics

Previous sections have shown that image dynamics can be rigid, affine or Riccati. Note also that motion and shape parameters enter into these dynamics linearly. In other words, we can rewrite these image dynamics as

$$\dot{\xi} = f(\xi)\varphi \tag{14}$$

where $\xi \in \mathbb{R}^2$ is the image feature vector, i.e. $\xi = (x, y)^T$, $f(\xi)$ is a linear (or nonlinear) mapping and φ is the vector

of unknown parameters which can be either constant or time-varying. Suppose we have n parameters to be estimated, i.e. $\varphi \in \mathbf{R}^n$ and $f \in \mathbf{R}^{2\times n}$. From (14) it is clear that each image feature provides 2 equations. For the estimation of n parameters, we will therefore require at least $m \triangleq n/2$ independent image features. For example, rigid motion (12) in 2D is defined by 3 parameters (ω, b_1, b_2) and we need at least 1.5 image points. In the case of affine motion (11), 3 image features imply 6 equations which can be solved for 6 affine parameters, $a_1, a_2, a_3, a_4, b_1, b_2$. In the case of Riccati dynamics (10), we need at least 4 image features since they will imply 8 equations which can be used to solve 8 parameters, $d_1, d_2, \ldots, d_8$.

Let us consider m image features concatenated in the following dynamical system:

$$\underbrace{\begin{pmatrix} \dot{\xi}_1 \\ \dot{\xi}_2 \\ \vdots \\ \dot{\xi}_m \end{pmatrix}}_{\dot{\Xi}} = \underbrace{\begin{pmatrix} f_1(\xi_1) \\ f_2(\xi_2) \\ \vdots \\ f_m(\xi_m) \end{pmatrix}}_{F} \varphi \Rightarrow \dot{\Xi} = F\varphi \tag{15}$$

where $\xi_r = \begin{pmatrix} x_r & y_r \end{pmatrix}^T$ and $r = 1, 2, \ldots, k$. Note that F is now a square matrix with $m \times m$. Since the image features are extracted from images, they are measurable and therefore we can also introduce an output (read-out) equation as

$$Y = \Xi \tag{16}$$

IV. Sliding Mode Observers

In this section, we construct an observer whose state follows the state of motion of an extracted image feature as closely as possible. We will employ sliding mode control (SMC) in the design of the observer to stabilize the state estimation error around zero. Basically we are copying our image feature dynamics and try to control this copied version by SMC. More precisely, let our observer be

$$\dot{\hat{\Xi}} = u \tag{17}$$

where u will be designed using SMC so that $\tilde{\Xi} = \Xi - \hat{\Xi} \to 0$ as $t \to \infty$.

Let us define the sliding mode manifold as

$$\sigma = Y - \hat{\Xi},$$

which then implies that

$$\dot{\sigma} = \dot{Y} - \dot{\hat{\Xi}} = F\varphi - u. \tag{18}$$

Let us pick the following Lyapunov function

$$V = \frac{1}{2}\sigma^T\sigma$$

whose time derivative is

$$\dot{V} = \sigma^T\dot{\sigma}$$

which can be made negative definite by setting $\dot{\sigma}$ to either $-MSgn(\sigma)$, where $M > 0$ and $Sgn(.)$ is the signum function, or $-D\sigma$, where D is a positive definite matrix. If $-MSgn(\sigma)$ is selected, all components of the control are switching between lower and upper bound of control. This may cause unnecessary chattering in the system especially in the discrete-time implementation of the control algorithm. Combination of the $\dot{\sigma} = -MSgn(\sigma)$ and $\dot{\sigma} = -D\sigma$ by selecting $\dot{\sigma} = -D\sigma - \rho(x,t)Sgn(\sigma)$ yields a solution that may combine good properties of both solutions and allows selecting $\rho(x,t)$ small enough to minimize chattering and at the same time to guarantee the existence of sliding mode.

So by selecting $\dot{\sigma} = -D\sigma - \rho(x,t)Sgn(\sigma)$, $\dot{V}$ then becomes

$$\dot{V} = -\sigma^T D\sigma - \rho(x,t)\sigma^T Sgn(\sigma) = -\sigma^T D\sigma - \rho(x,t)||\sigma||,$$

which is clearly negative definite since $D > 0$ and $\rho > 0$. Therefore for stability,

$$\dot{\sigma} = -D\sigma - \rho Sgn(\sigma) \Rightarrow \dot{\sigma} + D\sigma + \rho Sgn(\sigma) = 0 \tag{19}$$

must be satisfied.

As shown in [43], u can be computed recursively using the algebraic distance $\dot{\sigma} + D\sigma$, namely

$$u(k) = u(k-1) + (\dot{\sigma} + D\sigma)|_k$$
$$= u(k-1) + \{\frac{\sigma(kT) - \sigma((k-1)T)}{T} + D\sigma(kT)\} \tag{20}$$

or,

$$u(k) = u(k-1) + \frac{1}{T}\{(I + TD)\sigma(kT) - \sigma((k-1)T)\} \tag{21}$$

where I is the identity matrix and T is the sampling time.

In light of (19), we can modify the algebraic distance used in (IV) as $\dot{\sigma} + D\sigma + \rho Sgn(\sigma)$ and compute the control again recursively as:

$$\hat{u}(k) = \hat{u}(k-1) + (\dot{\sigma} + D\sigma + \rho Sgn(\sigma))|_k$$
$$= \hat{u}(k-1) + \frac{1}{T}\{(I+TD)\sigma(kT) - \sigma((k-1)T) + \rho Sgn(\sigma(kT))\} \tag{22}$$

Since F is square and invertible, plugging (18) and (22) in (19) implies

$$F\hat{\varphi} - \hat{u} + D\sigma + \rho Sgn(\sigma) = 0 \Rightarrow \hat{\varphi} = F^{-1}(\hat{u} - D\sigma - \rho Sgn(\sigma)). \tag{23}$$

Note that when $\dot{\sigma} + D\sigma + \rho Sgn(\sigma) \to 0$ then $\hat{u} \to \hat{u}_{eq}$, namely we have the *"equivalent control"*, $\hat{u}_{eq}$, on the sliding manifold. In other words, we are estimating the parameters using the equivalent control.

V. Simulation Results

The proposed approach is simulated using Matlab7 and Simulink 6.0. FR denotes the frame rate of the camera in fps (frame per second). Since image data is available in every 1/FR seconds (which is larger than simulation sample time), we constructed a filter that interpolates the state values between two consecutive frames. We considered two cases: high frame rate (200 fps) and low frame rate (30 fps). At 30 fps, we also needed a filter to get non oscillatory results. To simulate the behavior of the camera, we constructed motion

dynamics subsystem in Simulink, which generates state values, i.e. image coordinates of the object, at prescribed frame rates. This is achieved using zero-order hold at the output port of the motion dynamics subsystem with a sample time equal to 1/FR. Interpolater used is the well known $G(s) = \frac{w_n^2}{s^2+2\varsigma w_n s+w_n^2}$ with $w_n = 50, \varsigma = 1$, which creates smooth values of the states between two consecutive frames. Sample time of the simulations is 0.0001 sec. Run time is 10 sec. Controller employed is exactly the one given by eqn (22), with $D = 0.1$ and $\rho = 0.0001$. Estimated parameters are filtered at low frame rate simulations, with a first order filter of time constant $\tau = 0.05$.

A. Rigid Motion Estimation

Rigid motion with time varying parameters was simulated with 200 fps. As time varying functions, step, sine and square wave were used for the parameters. Motion trajectory of the object is depicted in Fig. 2. Some of actual and estimated values are superimposed in Fig.3.

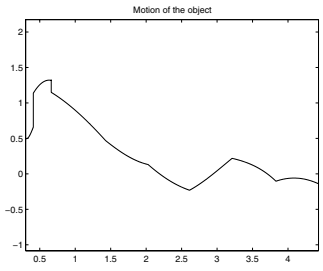
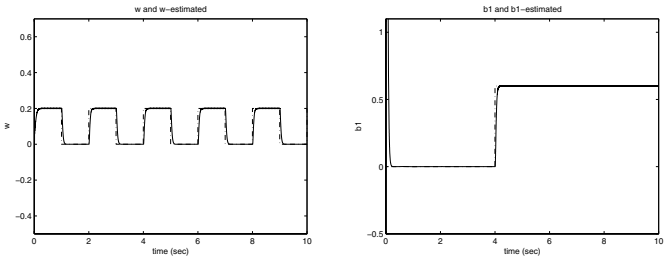


Fig. 2. Motion Trajectory



(a) **(b)**

Fig. 3. (a) ω (dashed) and $\hat{\omega}$ (solid), (b) b_1 (dashed) and $\hat{b}_1$ (solid)

There are two major differences between actual and estimated values. First we have a very small time delay; estimated values follow the actual ones with a small lag. This is due to the second order interpolater we use, since the interpolater needs previous and 2nd previous data to generate the current data. Also in the case of b_1 and ω, we don't see any sharp corners as in the actual ones, because interpolater kills these corners and produces a smoother response.

As can be seen from plotted graphs, our algorithm works fine, and fast estimation of the parameters with acceptable accuracy is achieved. Note that parameters converge to their actual values as soon as the sliding manifold is reached.

B. Affine Motion Estimation

Time varying parameter affine motion was simulated with 200 fps. Motion trajectory of the object is depicted in Fig. 4. Actual and estimated values are superimposed in Fig.5-Fig.7.

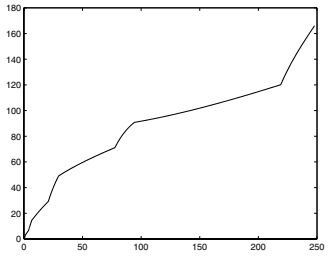
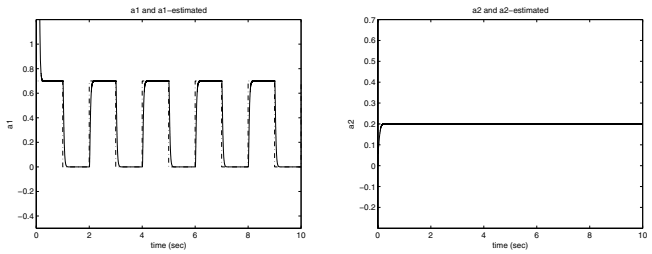
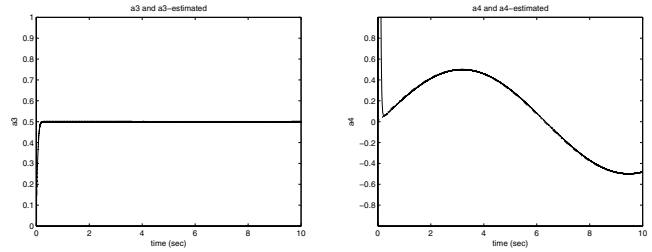


Fig. 4. Motion Trajectory



(a) **(b)**

Fig. 5. (a) a_1 (dashed) and $\hat{a}_1$ (solid), (b) a_2 (dashed) and $\hat{a}_2$ (solid)



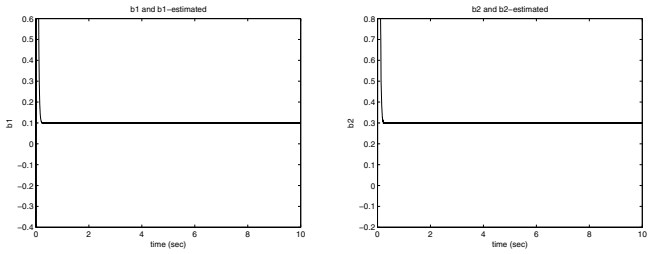
(a) **(b)**

Fig. 6. (a) a_3 (dashed) and $\hat{a}_3$ (solid), (b) a_4 (dashed) and $\hat{a}_4$ (solid)

VI. Experimental Results

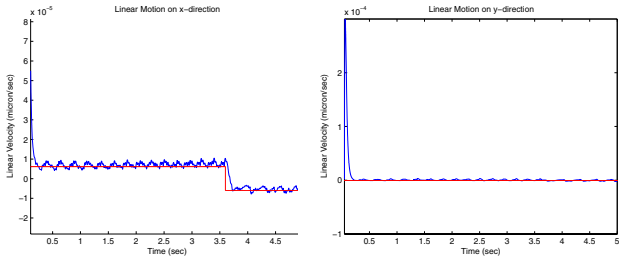
A. Experimental Setup of Vision System

Some initial real time experiments were conducted with the vision setup shown in Fig. 8. The system consists of Nikon SMZ 1500 Stereo Optical Microscope and a Firewire CCD camera with 30 fps on top of it. What is observed as a motion is created by NanoCube, which can provide nanometer range accuracy motion and positioning in XYZ directions. The actual control of Nanocube is done through dSPACE 1103 board and its control software. The points of interest to be tracked are chosen from micrometer, on which has a micron range



(a) **(b)**

Fig. 7. (a) b_1 (dashed) and $\hat{b}_1$ (solid), (b) b_2 (dashed) and $\hat{b}_2$ (solid)



(a) **(b)**

Fig. 9. (a) Actual (dashed) and Estimated (solid) linear velocity, b_1 along x axis, (b) Actual (dashed) and Estimated (solid) linear velocity, b_2 along y axis

checker board pattern, on top of NanoCube using some image processing operations in the OpenCV library. Each (1/30) second, image is taken, processed, and the points of interest are extracted from the image. To synchronize the sample time of capturing images and the sample time of the control, the same interpolation technique is used as in simulations.

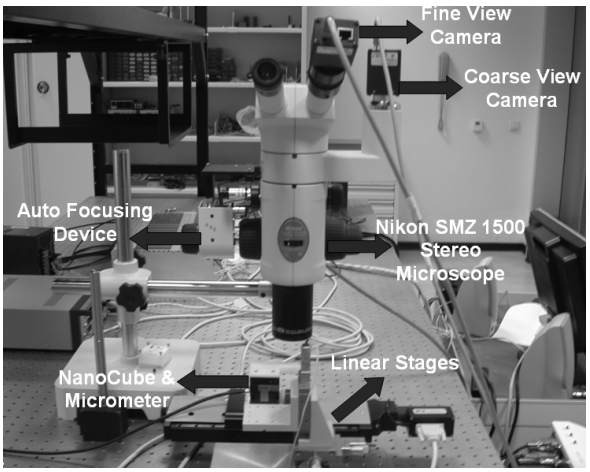


Fig. 8. Experimental Vision Setup

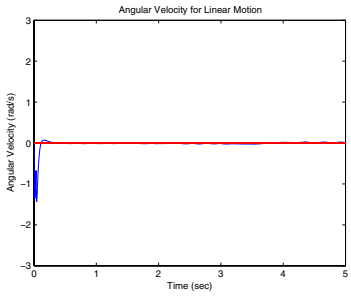


Fig. 10. Estimated angular velocity, ω

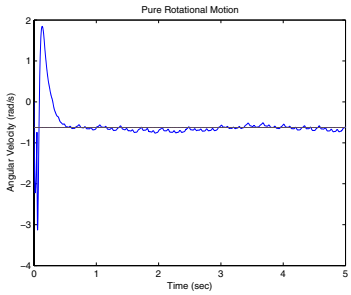


Fig. 11. Actual (dashed) and Estimated (solid) angular velocity, ω

B. Results for Rigid Motion

1) Linear Motion along x-axis: In this case NanoCube moves along x-axis with 6 μ m/s for a while and stops, and then turns back to opposite direction again with a 6 μ m/s. Since there is no motion along y-axis, the estimated linear velocity along y-axis is around zero. Furthermore, since there is no rotational motion, the estimated angular velocity is again around zero. (See Fig. 9-10.)

2) Pure Rotational Motion: In this case NanoCube undergoes a pure rotational motion with a constant angular velocity, $\pi/5$ rad/s in counterclockwise. The radius of the circular motion is 10 μm. Actual and estimated angular velocities are superimposed in Fig. 11.

Although, the reference motion is pure rotation, the angular velocity becomes very large for small time interval at very beginning. Then it converges to reference value very quickly. At the same time, the estimated trajectory seems to undergo linear motion (See Fig. 12). The reason is that the initial points, which are given as initial conditions for estimation, are different from the actual points on the circle. Therefore estimated points show some linear behavior until the dynamic of the system reaches the sliding manifold. When the system is on the sliding manifold, estimated values converge to the actual values and hence the motion becomes circular.

VII. Conclusions

We have developed a new framework to solve motion and structure estimation problems in machine vision using sliding mode observers. We have shown that most of the dynamics encountered in machine vision applications can be recast as dynamical systems which are linear in terms of parameters to be estimated. We then constructed an appropriate sliding observer whose output follows the output of the original

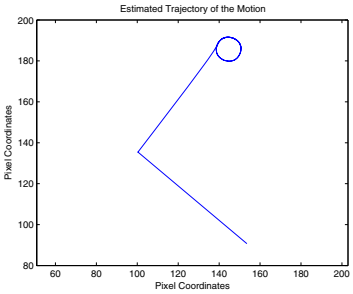


Fig. 12. Estimated Motion Trajectory

dynamics asymptotically. Control which drives the observer has been computed recursively based on the algebraic distance $\dot{\sigma} + D\sigma + \rho Sgn(\sigma)$.

Simulation and initial experimental results are promising and much has to be done to assess the performance of our proposed algorithm with various data perturbations.

REFERENCES

[1] A. Calway. Recursive Estimation of 3D Motion and Surface Structure from Local Affine Flow Parameters, IEEE Trans. on PAMI. vol. 27, no. 4 (2005) 562-574.

[2] B. K. P. Horn, Robot Vision, The MIT Press, 1986.

[3] G. Adiv. Determining three dimensional motion and structure from optical flow generated by several objects, IEEE Trans. on Pattern Anal. Mach. Intell. 7 (1985) 384-401.

[4] J. Aloimonos and C.M. Brown, Preception of structure from motion, I: Optical flow v.s. discrete displacements, in Proc. IEEE Conf. Computer Vision and Pattern Recognition, June, 1986.

[5] Y. Liu and T.S. Huang, Estimation of rigid body motion using straight line correspondence, Proc. Workshop on Motion: Representation and Analysis, IEEE Computer Society Press, 1986.

[6] Y. Liu and T.S. Huang. A linear algorithm for motion estimation using straight line correspondences, Computer Vision, Graphics, and Image Process, vol. 44, 1988.

[7] O.D. Faugeras, F. Lustman, and G. Toscani. Motion and structure from point and line matches, INRIA, BP105, Domain de Volucen, rocquencourt, 78153 Le Chesnay, France, 1987.

[8] O.D. Faugeras, On the motion of 3D curves and its relation to optical flow. Proc. First European Conference on Computer Vision (ed. O. Faugeras). Lecture Notes in Computer Science, vol. 427, pp. 107-117. Berlin, Heidelberg and New York: Springer Verlag, 1990.

[9] O.D. Faugeras and S. Maybank, Motion from point matches: Multiplicity of solutions, Int. J. Comput. Vision, vol. 4, pp. 225-246, 1990.

[10] A.M. Waxman and S. Ullman, Surface structure and 3-D motion from image flow: kinematic analysis, Int'l. J. Robotics Research 4 (1985) 72-94.

[11] S. Ullman, The interpretation of visual motion, Cambridge, MA : MIT Press, 1979.

[12] K.I. Kanatani. Detecting motion of a planar surface by line and surface integrals, Computer Vision, Graphics and Image Processing 29 (1985) 13-22.

[13] K. Kanatani, Group-Theoretical Methods in Image Understanding, Springer Verlag, 1990.

[14] A. Mitiche and J.K. Aggarwal. Analysis of time varying imagery, Handbook of pattern recognition and image processing, eds T.Y. Young and K.S. Fu, Academic Press 1986, pp. 311-332.

[15] M. E. Spetsakis and J. Aloimonos, Structure from motion using line correspondences, International J. Computer Vision, 4, pp. 171-183, 1990.

[16] M. E. Spetsakis and J. Aloimonos, Optimal computing of structure from motion using point correspondences in two frames, Proc. Second Int. Conf. Comput. Vision (Tampa, FL), Dec. 1988, pp. 449-453.

[17] M. E. Spetsakis and J. Aloimonos, A multiframe approach to visual motion perception, Int. J. Comput. Vision, vol. 6, pp. 245-255, 1991.

[18] J.W. Roach and J.K. Aggarwal, Determining the movement of objects from a sequence of images, IEEE Trans. on Pattern Anal. Mach. Intell. 6 pp. 554-562, 1980.

[19] H. H. Nagel, Representation of moving rigid objects based on visual observations, Comput., vol. 14, no. 8, pp. 29-39, 1981.

[20] H. C. Longuet-Higgins, A computer algorithm for reconstructing a scene from two projections, Nature, vol. 293, no. 10, pp. 133-135, 1981.

[21] R.Y. Tsai and T. S. Huang. Uniqueness and estimation of three dimensional motion parameters of rigid objects with curved surfaces, IEEE Transactions on pattern analysis and machine intelligence, vol. 6, no. 1, pp. 13-26, January 1984.

[22] X. Zhuang, T. S. Huang and R. M. Haralick, Two-view motion analysis: A unied algorithm, J. Opt. Soc. Amer., vol. A-3, pp. 1492-1500, 1986.

[23] X. Zhuang, A simplication to linear two-view motion algorithm, Comput. Vision Graphics Image Process., vol. 46, pp. 175-178, 1989.

[24] A. N. Netravali, T. S. Huang, A. S. Krishnakumar and R. J. Holt, Algebraic methods in 3-D motion estimation from two-view point correspondences, Int. J. Imaging Syst. Tech., vol. 1, pp. 78-99, 1989.

[25] C. Jerian and R. Jain, Polynomial methods for structure from motion, IEEE Trans. Patt. Anal. Machine Intell., vol. 12, pp. 1150-1165, 1990.

[26] C. Jerian and R. Jain, Structure from motion - A critical analysis of methods, IEEE Trans. Sys. Man Cybern., vol. 21, pp. 572-588, 1991.

[27] J. Weng, T. S. Huang and N. Ahuja, Motion and structure from two perspective views: Algorithms, error analysis and error estimation, IEEE Trans. Patt. Anal. Machine Intell., vol. 11, pp. 451-467, 1989.

[28] N. M. Grzywacz and E. C. Hildreth, Incremental rigidity scheme for recovering structure from motion: position-based versus velocitybased methods, J. Optical Soc. America, Series A 4, 503-518, 1987.

[29] D. W. Murray and B. F. Buxton, Experiments in the machine interpretation of visual motion, Cambridge, MA: The MIT Press, 1990.

[30] S. Maybank, Theory of reconstruction from image motion, Springer series in information sciences, 28, Springer Verlag, 1993.

[31] S. Amari. Feature spaces which admit and detect invariant signal transformation, in Proc. 4th Int'l Joint Conf. on Pattern Recognition, 1978, pp. 452-456.

[32] S. Amari. Invariant structures and feature spaces in pattern recognition problems, R.A.A.G.,Memoirs 4 (1968) 553-556.

[33] M. J. Black and A. Jepson, IEstimating Optical Flow in Segmented Images Using Variable-Order Parametric Models with Local Deformations, IEEE Trans. on PAMI. no. 10, (1996) 972-985.

[34] B. K. Ghosh, M. Jankovic and Y. T. Wu, Perspective Problems in System Theory and its Application to Machine Vision, Journal of Mathematical Systems, Estimation and Control, vol. 4, No. 1, (1994), pp. 3-38.

[35] B. K. Ghosh, E. P. Loucks, A Perspective Theory for Motion and Shape Estimation in Machine Vision, SIAM Journal of Control and Optimization, vol. 33, No. 5, pp. 1530-1559, September 1995.

[36] B. K. Ghosh, E. P. Loucks, A Realization Theory for Perspective Systems with Applications to Parameter Estimation Problems in Machine Vision, IEEE Transactions on Automatic Control, vol. 41, No. 12, 1996.

[37] J. K. Aggarwal and N. Nandhakumar, On the Computation of Motion from Sequences of Images: A Review, Proc. IEEE, vol. 76, No. 8, Aug. 1988.

[38] B. K. P. Horn and M. J. Brooks. Shape from shading, The MIT Press, Cambridge, MA, 1989.

[39] Canny, J., A Computational Approach to Edge Detection, IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 8, pp. 679-698, 1986.

[40] Marr, D., Vision, W. H. Freeman and Compaany, 1982.

[41] Rosenfeld, A. and Thurston, M., Edge and curve detection for visual scene analysis, IEEE Trans. on Computers, 1971, vol. C-20.

[42] V. Utkin, Sliding Modes in Control Optimization, Springer-Verlag, 1992.

[43] K. Abidi, A. Sabanovic, A Study on Discrete Sliding Mode Control: Robustness Analysis and Experimental Investigation, Proceedings of IASTED International Conference on Modelling, Identification and Control (MIC 2005), Innsbruck, Austria, February 16-18, 2005.