Proceedings of the 39th IEEE
Conference on Decision and Control
Sydney, Australia • December, 2000

Robust Control and Tracking

Allen Tannenbaum
Department of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA 30082

Abstract

In this note, we consider some problems in active vision for which techniques in robust control may be very relevant. In particular, we discuss eye tracking problems. We also survey some key ideas from active vision including optical flow and deformable contours.

1 Introduction

In this paper, we consider some general ideas for a theory of controlled active vision. We will use as a model problem that of tracking, in particular tracking eye movements. We will indicate that one can treat this problem by using adaptive and robust control in conjunction with multiscale methods from signal processing, and shape recognition theory from computer vision. Tracking is a basic control problem in which we want the output to follow or track a reference signal, or equivalently we want to make the *tracking error* as small as possible relative to some well-defined criterion (say energy, power, peak value, etc.). Even though tracking in the presence of a disturbance is a classical control issue, the problem at hand is very difficult and challenging because of the highly uncertain nature of the disturbance. There are a number of tracking problems that can easily be considered in a university environment and which could act as benchmarks for testing various algorithms. For example, one could consider the eye movement tracking problem in the context of a man-computer interface.

The techniques which we will discuss below should have a wide range of applicability in a number of tracking problems including those in robotics, remotely controlled vehicles, and pilot tracking helmets currently being developed. The latter are systems combining helmet mounted head and eye track capability to define a subject's true line of sight. Clearly the proper exploitation of the dynamic characteristics of the human visual system by tracking the postion of the viewer's eyes leads to drastic reduction in the amount of information that needs to be transmitted.

We should note that the problem of visual tracking differs from standard tracking problems in that the feedback signal is measured using imaging sensors. In particular, it has to be extracted via computer vision algorithms and interpreted by a reasoning algorithm before being used in the control loop. Furthermore, the response speed is a critical aspect. For obvious reasons, an eye tracking system should be as non-invasive as possible. The eye movement can be tracked by studying images acquired by grey scale or infra-red cameras or by an array of sensors. The images are analyzed to extract the relative motions of the eyes and the head. The low level data acquisition and recognition could be accomplished using multiscale technique. Recognition could also be accomplished using such a multiscale approach and a new computational theory of shape. Consequently, from the control point of view, we have a tracking problem in the presence of a highly uncertain disturbance which we want to attenuate. Note that the uncertainty is due to the sensor noise (classical), the algorithmic component described above (uncertainty in extracted features, likelihood of various hypotheses, etc.), and modelling uncertainty.

2 Visual Tracking

There have a number of papers on the use of vision in tracking, especially in the robotics community. (See e.g., [4], [13], and the references therein.) Typically, the issue addressed in this work is the use of vision for servoing a manipulator for tracking (or an equivalent problem). The motivation for using vision in such a framework is clear, that is, the combination of computer vision coupled with control can be employed to improve the measurements. Indeed, because of improvements in image processing techniques and hardware, robotic technology is reaching the point where vision information may become an integral part of the feedback signal. For problems with little uncertainty, simple PID controllers have been used, and for more noisy systems, adaptive schemes as well as stochastic based LQG controllers have been utilized.

A number of control schemes have been proposed for the utilization of visual information in control loops. These have ranged from the use of sensory features to characterize hierarchical control structures, Fourier descriptors, and image segmentation. Approaches based on *optical flow* have been used as a key element in the calculation of the robot's driving signals (see in particular, [13], and our discussion below). Indeed, since an object in an image is made up of brightness patterns, as the object moves in space so do the brightness patterns. The optical flow is then the apparent motion of the brightness patterns; see [6]. Under standard assumptions, given a static target and a moving camera (or equivalently, a static object and a moving camera), one can write down equations for the velocity of a projected point of the object onto the image plane. Several methods have been discussed for the computations of this velocity. This information can then be used to track the image.

Let us consider the concrete problem of tracking targets on a computer screen. Then we have only a two dimensional tracking question. We assume for simplicity that the object moves in a plane which is perpendicular to the optical axis of the camera. If the camera then moves with translation velocity

$$\tau := \begin{bmatrix} \tau_x \\ \tau_y \end{bmatrix},$$

and rotational velocity ρ_z (with respect to the camera frame), one may pose the two dimensional tracking problem as follows for an object [14]. Let P_t denote the area on the image plane which is the projection of the target. Then visually tracking this feature amounts to finding the camera translation τ and rotation ρ_z (with respect to the camera frame) which keeps P_t stationary. There are similar characterizations for the tracking of features. Now from this set-up, one can write down linearized equations of the optical flows generated by the motion of the camera where the control variables are given by those components of the optical flow induced by the camera's tracking motion.

The exact form may be found in [14], and need not concern us now. The point is that the resulting system may be written in standard state space form and after discretization (with T the time between two consecutive frames) takes on the form

$$\begin{aligned} x(n+1) &= x(n) + Tu_r(n) + Td(n) + v(n), \\ z(n) &= x(n) + w(n), \end{aligned}$$

where u_r is the reference, d is the exogenous disturbance, v is a "noise" term for the model uncertainty, z is the measurement together with noise component w. (All the vectors are in $\mathbf{R}^3$. The components of the state vector are made up of the the x, y, and roll component of the tracking error.)

There are a number of important control issues related to such a set-up. Of course, one has the problem of measurement delays (we want to work in real time) and choice of sampling time. But we feel there is a much deeper and more difficult problem which must be addressed before a reasonable choice of control strategy can be made. Namely, in general the uncertainty (v and w) is modelled as white noise. This model is conservative and does not bring into account any of the possible structure of noise environment. One of the key contributions in modern robust control has the consideration of *structure* in uncertainty.

In our case, we are proposing a much deeper analysis of the uncertainty connected to such problems. This brings the key element of *signal processing* and in particular, the new powerful methods of *multiscale computations*. *Shape recognition theory* in computer vision based on Hamilton-Jacobi theory will also play a key role in this program as will be argued below.

3 Image Feature Extraction

In the context of eye tracking, there are two basic approaches to data acquisition: an active approach and a passive imaging technique.

In the active approach, a harmless near-infrared light is used to illuminate the user's face. Two reflections from the eyes are then extracted. The first reflected signal is due to the corneal surface and is called the *glint*. The second reflection occurs off the retina and is called the *bright eye component*. To minimize the effect of background radiation, current systems typically require a dim lighting.

The passive approach to data acquisition is again based on extracting the two reflections from an infra-red image of the scene. In this sense the approach is similar to the active system. The passive system that we consider simply consists of a grey scale levels camera. The camera is covered with light filters to attenuate diffuse reflections due to background illumination and enhance the pupil.

3.0.1 Contour Map: In both cases, our immediate objective is to either extract the location of the pupil and a feature of the face such as the nose, or extract the glint and bright eye reflections from the image. We can accomplish this by first producing an *edge* image using a snakes based approach. In this section, we will describe a new paradigm for *snakes* or *active contours* based on principles from curvature driven flows which because of its speed and accuracy seems ideally suited for edge extraction in this context.

Active contours may be regarded as autonomous processes which employ image coherence in order to track various features of interest over time. Such deformable contours have the ability to conform to various object shapes and motions. Snakes have been utilized for segmentation, edge detection, shape modelling, and visual tracking.

In the classical theory of snakes, one considers energy minimization methods where controlled continuity splines are allowed to move under the influence of external image dependent forces, internal forces, and certain contraints set by the user. As is well-known there may be a number of problems associated with this approach such as initializations, existence of multiple minima, and the selection of the elasticity parameters. Moreover, natural criteria for the splitting and merging of contours (or for the treatment of multiple contours) are not readily available in this framework.

In [7], we propose a novel deformable contour model to successfully solve such problems, and which will become one of our key techniques for tracking. Our method is based on the Euclidean curve shortening evolution (see our discussion below) which defines the gradient direction in which a given curve is shrinking as fast as possible relative to Euclidean arc-length, and on the theory of conformal metrics. We multiply the Euclidean arc-length by a conformal factor defined by the features of interest which we want to extract, and then we compute the corresponding gradient evolution equations. The features which we want to capture therefore lie at the bottom of a potential well to which the initial contour will flow. Moreover, our model may be easily extended to extract 3D contours based on motion by mean curvature [7].

Let us briefly review some of the details from [7]. First of all, in [3, 10] a snake model based on the level set formulation of the Euclidean curve shortening equation is proposed. More precisely, the model is

$$\frac{\partial \Psi}{\partial t} = \phi(x,y)||\nabla\Psi||(\mathrm{div}(\frac{\nabla\Psi}{||\nabla\Psi||}) + \nu). \qquad (1)$$

Here the function $\phi(x,y)$ depends on the given image and is used as a "stopping term." For example, the term $\phi(x,y)$ may chosen to be small near an edge, and so acts to stop the evolution when the contour gets close to an edge. One may take [3, 10]

$$\phi := \frac{1}{1+||\nabla G_\sigma * I||^2}, \qquad (2)$$

where I is the (grey-scale) image and G_σ is a Gaussian (smoothing filter) filter. The function $\Psi(x,y,t)$ evolves in (1) according to the associated level set flow for planar curve evolution in the normal direction with speed a function of curvature which was introduced in [12]. It is important to note that the Euclidean curve shortening part of this evolution, namely

$$\frac{\partial \Psi}{\partial t} = ||\nabla\Psi||\mathrm{div}(\frac{\nabla\Psi}{||\nabla\Psi||}) \qquad (3)$$

is derived as a gradient flow for shrinking the perimeter as quickly as possible. As is explained in [3], the constant *inflation term* ν is added in (1) in order to keep the evolution moving in the proper direction.

We would like to modify the model (1) in a manner suggested by the curve shortening flow. We change the ordinary arc-length function along a curve $C = (x(p), y(p))^T$ with parameter p given by

$$ds = (x_p^2 + y_p^2)^{1/2}dp,$$

to

$$ds_\phi = (x_p^2 + y_p^2)^{1/2}\phi dp,$$

where $\phi(x,y)$ is a positive differentiable function. Then we want to compute the corresponding gradient flow for shortening length relative to the new metric ds_ϕ.

Accordingly set

$$L_\phi(t) := \int_0^1 ||\frac{\partial C}{\partial p}||\phi dp.$$

Then taking the first variation of the modified length function L_ϕ, and using integration by parts (see [7]), we get that

$$L'_\phi(t) \quad = \quad -\int_0^{L_\phi(t)} \langle \frac{\partial C}{\partial t}, \phi\kappa\vec{\mathcal{N}} - (\nabla\phi\cdot\vec{\mathcal{N}})\vec{\mathcal{N}}\rangle ds$$

which means that the direction in which the L_ϕ perimeter is shrinking as fast as possible is given by

$$\frac{\partial C}{\partial t} = (\phi\kappa - (\nabla\phi\cdot\vec{\mathcal{N}}))\vec{\mathcal{N}}. \qquad (4)$$

This is precisely the gradient flow corresponding to the miminization of the length functional L_ϕ. The level set version of this is

$$\frac{\partial \Psi}{\partial t} = \phi||\nabla\Psi||\mathrm{div}(\frac{\nabla\Psi}{||\nabla\Psi||}) + \nabla\phi\cdot\nabla\Psi. \qquad (5)$$

One expects that this evolution should attract the contour very quickly to the feature which lies at the bottom of the potential well described by the gradient flow (5). As is standard, we may also add a constant inflation term, and so derive a modified model of (1) given by

$$\frac{\partial \Psi}{\partial t} = \phi||\nabla\Psi||(\mathrm{div}(\frac{\nabla\Psi}{||\nabla\Psi||}) + \nu) + \nabla\phi\cdot\nabla\Psi. \qquad (6)$$

Notice that for ϕ as in (2), $\nabla\phi$ will look like a doublet near an edge. Of course, one may choose other candidates for ϕ in order to pick out other features. Clearly, the ability of these snakes to change topology, and quickly capture the desired features can make them an pwoerful tool for tracking algorithms.

More specifically, let us review briefly how snakes may be used for tracking. One uses a deformable contour to capture the given object at time $t-1$ in a video sequence of images, and then uses the final position of the latter contour as the initial position of the active contour at time t. The success of the method is dependent on having the deformable contour being in the basin of attraction of the object at time t. In many cases an estimation step is necessary for which one may employ a Kalman filter ("Kalman snakes" [1]). Also other visual cues may be necessary such as those provided by optical flow to be described below.

3.0.2 Contour Shape Identification: The edge map that we produce via the multiscale approach is then analyzed to identify the shape of each contour. This allows us to associate the contour with one of the features (e.g. glint or bright eye reflection, nose contour, etc.) that we use to infer the instantaneous eye movement. We plan to exploit the our work in shape recognition and decomposition in our research [8].

In this work, a new approach is given for a computational theory of shape which combines morphological operations with Gaussian smoothing in one framework. Moreover, an innovative notion of scale for signals can be applied to the problem of shape extraction. Our work can also be regarded as a generalization of certain useful morphological filters, and so offers a *nonlinear multiscale* framework for feature and motion detection in computer vision. See also [2, 11].

The approach is based on a reaction-diffusion equation derived from the curve evolution

$$\begin{aligned} \frac{\partial C}{\partial t} &= (\alpha + \beta\kappa)\vec{\mathcal{N}}, \qquad (7) \\ C(\cdot, 0) &= C_0(\cdot), \end{aligned}$$

where $\vec{\mathcal{N}} = \vec{\mathcal{N}}(\cdot, t)$ denotes the unit normal and $\kappa = \kappa(\cdot, t)$ the curvature of the curve $C(\cdot, t)$ of the family. (See [8] for all the details.) We can study the evolution of shapes under very general deformations, and show that they compose into two types, a deformation that is constant along the normal (morphology) and corresponds to a nonlinear hyperbolic wave type of process, and a deformation which varies with the curvature and corresponds to a quasi-linear diffusive one (a quasi-linear analogue of Gaussian smoothing). Indeed, this last type of deformation may be derived from the geometric heat equation or Euclidean curve shortening process. Its smoothing properties have been shown to be superior to those of conventional (linear) Gaussian smoothing in vision applications [8]. Our evolutions give rise to shocks, the singularities of shape, which provide a hierarchical decomposition of a shape into a given set of shape elements, e.g., parts and protrusions.

In fact, this approach gives a rigorous treatment of singularities in the context of shape theory, and playing off the parameters α and β against one another leads to a new *reaction-diffusion scale space*. Thus it allows one to formulate a novel framework for multi-scale filtering. Further, there is a strong connection to Hamilton-Jacobi theory in this research. See also [12], and [15].

To illustrate some of the above remarks, let us consider, the case in which $\alpha = 1, \beta = 0$. In this case, Equation (7) is a nonlinear hyperbolic equation, and so its solutions develop singularities, and thus a notion of *weak entropy* solution must be developed. This leads to the corresponding *viscosity* solution in the Hamilton-Jacobi formulation of the evolution [12]. Indeed, in this case, it is very easy to see how singularities may develop from such an evolution. One can easily show that in general for the evolution (7), if v denotes the arc-length parameter, then the curvature evolves as

$$\frac{\partial \kappa}{\partial t} = \beta\kappa_{vv} + \beta\kappa^3 - \alpha\kappa^2.$$

For $\alpha = 1, \beta = 0$, we get that

$$\frac{\partial \kappa}{\partial t} = -\kappa^2$$

whose solution is

$$\kappa(v, t) = \frac{\kappa(v, 0)}{1 + t\kappa(v, 0)}.$$

Thus if the curve has a point of negative curvature, the solution must become singular in finite time.

4 Tracking and Optical Flow

Once the contours corresponding to the various features that we wish to track have been identified, we use an optical flow procedure to estimate the motion of each feature from two consecutive images that contain only that feature. The images are obtained from the current and previous contour images by deleting all contours except for the one of interest. The resulting motion estimation problem is much better conditioned than the traditional optical flow estimation from pure intensity images. As we discussed above, optical flow field is the velocity vector field of apparent motion of brightness patterns in a sequence of images [6]. One assumes that the motion of the brightness patterns is the result of relative motion, large enough to register a change in the spatial distribution of intensities on the images. Thus, relative motion between an object and a camera can give rise to optical flow. Similarly, relative motion among objects in a scene being imaged by a static camera can give rise to optical flow.

In our computation of the optical flow we use work on generalized *viscosity* solutions to Hamilton-Jacobi type equations. Indeed, these techniques seem ideally suited for the variational Euler-Lagrange approaches to this problem (see also [5, 6, 16] and the references therein). Utilizing such generalized solutions, we have been able to handle the singularities and regularity problems for several distinct variational formulations of the optical flow that occur in this area.

4.1 L^1 Based Optical Flow

In [9], we consider a spatiotemporal differentiation method for optical flow. Even though in such an approach, the optical flow typically estimates only the isobrightness contours, it has been observed that if the motion gives rise to sufficiently large intensity gradients in the images, then the optical flow field can be used as an approximation to the real velocity field and the computed optical flow can be used reliably in the solutions of a large number of problems; see [6] and the references therein.

The problem of computing optical flow is ill-posed and so well-posedness has to be imposed by assuming suitable *a priori* knowledge. In [9], we employ a variational formulation for imposing such *a priori* knowledge. One constraint which has often been used in the literature is the "optical flow constraint"(OFC). The OFC is a result of the simplifying assumption of constancy of the intensity, $E = E(x, y, t)$, at any point in the image [6]. It can be expressed as the following linear equation in the unknown variables u and v

$$E_x u + E_y v + E_t = 0, \tag{8}$$

where E_x, E_y and E_t are the intensity gradients in the x, y, and the temporal directions respectively, and u and v are the x and y velocity components of the apparent motion of brightness patterns in the images, respectively. It has been shown that the OFC holds provided the scene has Lambertian surfaces and is illuminated by either a uniform or an isotropic light source, the 3-D motion is translational, the optical system is calibrated and the patterns in the scene are locally rigid.

It is not difficult to see from equation (8) that computation of optical flow is unique only up to computation of the flow along the intensity gradient $\nabla E = (E_x, E_y)^T$ at a point in the image [6]. This is the celebrated *aperture problem.* One way of treating the aperture problem is through the use of regularization in computation of optical flow, and consequently the choice of an appropriate constraint. A natural choice for such a constraint is the imposition of some measure of consistency on the flow vectors situated close to one another on the image.

In their pioneering work, Horn and Schunk [6] use a quadratic smoothness constraint. The immediate difficulty with this method is that at the object boundaries, where it is natural to expect discontinuities in the flow, such a smoothness constraint will have difficulty capturing the optical flow. For instance, in the case o f a quadratic constraint in the form of the square of the norm of the gradient of the optical flow field [6], the Euler-Lagrange (partial) differential equations for the velocity components turn out to be *linear* elliptic. The corresponding parabolic equations therefore have a linear diffusive nature, and tend to blur the edges of a given image. In the past, work has been done to try to suppress such a constraint in directions orthogonal to the occluding boundaries in an effort to capture discontinuities in image intensities that arise on the edges.

We have [9], a novel method for computing optical flow based on the theory of the evolution of curves and surfaces. The approach employs an L^1 type minimization of the norm of the gradient of the optical flow vector rather than quadratic minimization as has been undertaken in most previous regularization approaches. The equations that arise are nonlinear degenerate parabolic equations. The equations diffuse in a direction orthogonal to the intensity gradients, i.e., in a direction along the edges. This results in the edges being preserved. of the equations leads to solutions which incorporate the nature of the discontinuities in image intensities into the optical flow.

We can summarize our procedure as follows:

1. Let $E = E(x, y, t)$ be the intensity of the given moving image. Assume constancy of intensity at any point in the image, i.e.,

$$E_x u + E_y v + E_t = 0,$$

   where

$$u = \frac{dx}{dt}, \quad v = \frac{dy}{dt},$$

   are the components of the apparent motion of brightness patterns in the image which we want to estimate.

2. Consider the regularization of optical flow using the L^1 cost functional

$$\min_{(u,v)} \int\int \sqrt{u_x^2 + u_y^2} + \sqrt{v_x^2 + v_y^2} + \alpha^2 (E_x u + E_y v + E_t)^2 dxdy,$$

   where α is the smoothness parameter.

3. The corresponding Euler-Lagrange equations may be computed to be

$$\begin{aligned} \kappa_u - \alpha^2 E_x (E_x u + E_y v + E_t) &= 0, \\ \kappa_v - \alpha^2 E_x (E_x u + E_y v + E_t) &= 0, \end{aligned}$$

   where the curvature

$$\kappa_u := \operatorname{div}(\frac{\nabla u}{\|\nabla u\|}),$$

   and similarly for κ_v.

4. These equations are solved via "gradient descent" by introducing the system of nonlinear parabolic equations

$$\begin{aligned}\hat{u}_{t'} &= \kappa_{\hat{u}} - \alpha^2 E_x(E_x\hat{u} + E_y\hat{v} + E_t),\\ \hat{v}_{t'} &= \kappa_{\hat{v}} - \alpha^2 E_x(E_x\hat{u} + E_y\hat{v} + E_t),\end{aligned}$$

for $\hat{u} = \hat{u}(x, y, t')$, and similarly for $\hat{v}$.

The above equations have a significant advantage over the classical Horn-Schunck quadratic optimization method since they *do not blur edges.* Indeed, the diffusion equation

$$\begin{aligned}\Phi_t &= \Delta\Phi - \frac{1}{\|\nabla\Phi\|^2}\langle\nabla^2\Phi(\nabla\Phi), \nabla\Phi\rangle,\\ &= \kappa_\Phi\|\nabla\Phi\|\end{aligned}$$

does not diffuse in the direction of the gradient $\nabla\Phi$. Our optical flow equations are perturbations of the following type of equation:

$$\Phi_t = \frac{\kappa_\Phi}{\|\nabla\Phi\|}\|\nabla\Phi\|.$$

Since $\|\nabla\Phi\|$ is maximal at an edge, our optical flow equations do indeed preserve the edges. Thus the L^1–norm optimization procedure allows us to retain edges in the computation of the optical flow.

This approach to the estimation of motion will be one of the tools which we will employ in our tracking algorithms. The algorithm has already proven to be very reliable for various type of imagery [9].

5 Conclusions

In this paper, we sketched several techniques for treating tracking problems. We note that because of the uncertainty in the models and the signals, this class of problems presents a unique opportunity to researchers in robust control. Our approach is based on a combination of robust control, computer vision, and multiscale signal processing.

Recently, there has been a cross-fertilization among researchers in vision and control. Much of vision research until now has been open loop. When the loop has been closed very elementary control algorithms have been applied which have worked with mixed results. The use of visual information in a feedback loop can provide a rich new source of questions which has the potential of stimulating whole new areas of control research. The eye tracking problem outlined in this paper can be taken as a paradigm for a whole range of issues in controlled active vision. Indeed, the problem poses a powerful challenge to the robust control community.

References

[1] A. Blake and A. Yuille, *Active Vision*, MIT Press, Cambridge, MA, 1992.

[2] R. W. Brockett and P. Maragos, "Evolution equations for continuous-scale morphology," *Proc. ICASSP*, 1992, Volume III, 125-128.

[3] V. Caselles, F. Catte, T. Coll, and F. Dibos, "A geomteric model for active contours in image processing," *Numerische Mathematik* **66** (1993), pp. 1-31.

[4] J. T. Feddema, C. S. Lee, and O. R. Mitchell, "Weighted selection of image features for resolved rate visual feedback control," *IEEE Trans. Robotics and Automation* **7** (1991), 31-47.

[5] E. C. Hildreth, "Computations underlying the measurement of visual motion," *Artificial Intelligence* **23** (1984), 309-354.

[6] B. K. P. Horn, *Robot Vision,* MIT Press, Cambridge, Mass., 1986.

[7] S. Kichenassamy, A. Kumar, P. Olver, A. Tannenbaum, and A. Yezzi, "Conformal curvature flows: from phase transitions to active vision," *Archive for Rational Mechanics and Analysis* **134** (1996), pp. 275–301.

[8] B. B. Kimia, A. Tannenbaum, and S. W. Zucker, "Shapes, shocks, and deformations, I," *Int. J. Computer Vision* **15** (1995), pp. 189-224.

[9] A. Kumar, A. Tannenbaum, and G. Balas, "Optical flow: a curve evolution approach," *IEEE Transactions on Image Processing* **5** (1996), pp. 598–611.

[10] R. Malladi, J. Sethian, B. and Vermuri, "Shape modelling with front propagation: a level set approach," *IEEE PAMI* **17** (1995), pp. 158–175.

[11] P. Maragos, "Pattern spectrum and multiscale shape representation," *IEEE PAMI* **11** (1989), 701-716.

[12] S. J. Osher and J. A. Sethian, "Fronts propagation with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulations," *Journal of Computational Physics* **79** (1988), pp. 12-49.

[13] N. Papanikolopoulos, P. K. Khosla, and T. Kanade, "Vision and control techniques for robotic visual tracking," *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 1991.

[14] N. Papanikolopoulos, P. K. Khosla, and T. Kanade, "Adaptive robotic visual tracking," *Proceedings of American Control Conference*, Boston, MA, 1991.

[15] G. Sapiro and A. Tannenbaum, "Invariant curve evolution and image analysis," *Indiana University J. of Mathematics* **42** (1993), pp. 985–1009.

[16] M. Snyder, "On the mathematical foundations of smoothness constraints for the determination of optical flow and for surface reconstruction," *PAMI* **13**, pp. 1105-1114, 1991.