SAND REPORT

SAND2004-1869
Unlimited Release
Printed October 2004

Extracting Primary Features of a Statistical Pressure Snake

Hanspeter Schaub, ORION International Technologies

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation,
a Lockheed Martin Company, for the United States Department of Energy's
National Nuclear Security Administration under Contract DE-AC04-94-AL85000.

Approved for public release; further dissemination unlimited.

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone:	(865) 576-8401
Facsimile:	(865) 576-5728
E-Mail:	reports@adonis.osti.gov
Online ordering:	http://www.doe.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd
Springfield, VA 22161

Telephone:	(800) 553-6847
Facsimile:	(703) 605-6900
E-Mail:	orders@ntis.fedworld.gov
Online ordering:	http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online

SAND2004-1869
Unlimited Release
Printed October 2004

Extracting Primary Features of a Statistical Pressure Snake

Hanspeter Schaub
ORION International Technologies
Albuquerque, NM 87106
schaub@vt.edu

Abstract

Assume a target motion is visible in the video signal. Statistical pressure snakes are used to track a target specified by a single or a multitude of colors. These snakes define the target contour through a series of image plane coordinate points. This report outlines how to compute certain target degrees of freedom. The image contour can be used to efficiently compute the area moments of the target, which in return will yield the target center of mass, as well as the orientation of the target principle axes. If the target has a known shape such as begin rectangular or circular, then the dimensions of this shape can be estimated in units of image pixels. If the physical target dimensions are known apriori, then the measured target dimensions can be used to estimate the target depth.

Intentionally Left Blank

Contents

Figures

Intentionally Left Blank

Extracting Primary Features of a Statistical Pressure Snake

Introduction

Statistical pressure snakes are a relatively new means to segment an image. After defining a target color,[1] control points on a closed curve (snake) are moved such that they will settle down on the boundary of the target color. Assuming the target consists of a single color (or shades thereof), then the snake provides a parametric curve outlining the target outer edge.

This technical note discusses how this parametric curve can be used to extract target shape features such as the center of the shape, the primary shape axis orientation and the primary shape dimensions.

Calculating the Moments of a Parametric Curve

To extract the desired image shape features, it is required to compute the area moments M_{ij} of the parameterized curve. Given these moments, it is possible to compute the shape area, center of mass (C.M.), as well as the shape eigenfactors. Assume the snake is roughly outlining a shape as shown in Figure 1. The snake control points are shown as blue points, while the snake curve is a purple line. The target shape here is shown to be a box.

The area ij-th area moments of the shape are defined as

$$M_{ij} = \iint x^i y^j \,\mathrm{d}x\,\mathrm{d}y \tag{1}$$

where the area integral should be taken over the area defined by the closed snake curve. If the shaded area A in Figure 1 were defined through a set of pixel coordinates (x_n, y_m), then it would be trivial to compute the various area moments through

$$M_{ij} = \sum_{n=1}^{N} \sum_{m=1}^{M} x_n^i y_m^j \tag{2}$$

However, in the current problem setup the area A is defined through a parametric curve outlining this area. Recall Green's theorem which relates an area integral to a line integral

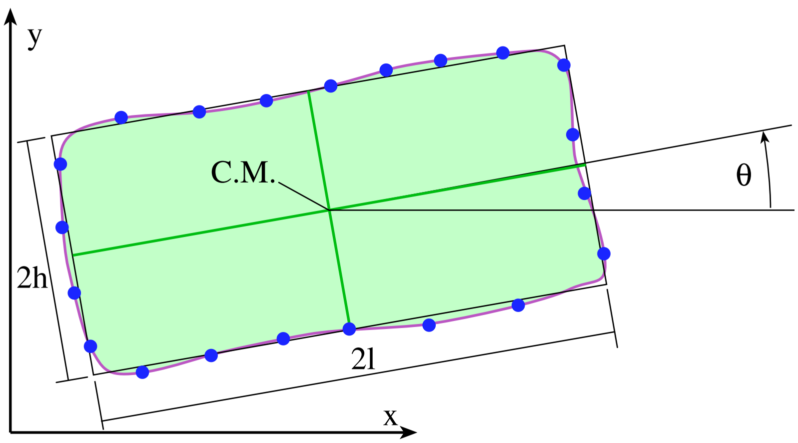

Figure 1. Illustration of a Snake Outlining a Target Shape

through

$$\iint \left(\frac{\partial Q}{\partial x} - \frac{\partial P}{\partial y} \right) \mathrm{d}x\,\mathrm{d}y = \oint P\mathrm{d}x + Q\mathrm{d}y \tag{3}$$

Using this theorem, it is possible to avoid the lengthy area integral and simply compute a line integral along the snake points. Reference 2 provides a well-written technical report which outlines how to take a closed polygon curve, defined through a set of discrete control points, and compute the various shape moments. The report provides the algorithms for moments up to second order. In the current snake implementation, this routine was not programmed, however. Instead the Open CV library was used which had an optimized routine built-in which computes the moments of an area defined through a discrete set of edge points.

Note that the area A of the target shape is simple the 00-th area moment:

$$A = M_{00} = \iint \mathrm{d}x\,\mathrm{d}y \tag{4}$$

The center of mass (x_c, y_c) of the shape is computed using the first area moments M_{10} and M_{01}:

$$x_c = \frac{M_{10}}{A} = \frac{1}{A} \iint x\,\mathrm{d}x\,\mathrm{d}y \tag{5a}$$

$$y_c = \frac{M_{01}}{A} = \frac{1}{A} \iint y\,\mathrm{d}x\,\mathrm{d}y \tag{5b}$$

Since an area integral is used to compute the target C.M., note that this computation is rather insensitive to noise and minor deviations of the snake contour to the actual target contour. Performing the integration, minor deviations along the snake typically cancel each other and yield a steady and accurate C.M. estimate.

Computing the Shape Eigenfactors

Given the target shape area and C.M., we would like to determine the principal shape orientation and size. These features are computed by evaluating the shape 2nd order moments. Here we assume that the coordinate system origin has been translated to coincide with the computed target shape center of mass. Let the inertia-like matrix $[I]$ be defined as

$$[I] = \begin{bmatrix} M_{20} & M_{11} \\ M_{11} & M_{02} \end{bmatrix} = \begin{bmatrix} I_{xx} & I_{xy} \\ I_{xy} & I_{yy} \end{bmatrix} \tag{6}$$

Note that $[I]$ is symmetric and positive definite. The principal orientation angle θ determines a coordinate system rotation which will yield a diagonal inertia matrix $[I']$. Let $[C]$ be the direction cosine matrix (rotation matrix) which will map the coordinate axis orientations between the two coordinate frames.

$$[C] = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \tag{7}$$

The rotation matrix is defined such that the coordinate transformation

$$[C][I][C]^T = [I'] \tag{8}$$

or

$$\begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} I_{xx} & I_{xy} \\ I_{xy} & I_{yy} \end{bmatrix} = \underbrace{\begin{bmatrix} I_1 & 0 \\ 0 & I_2 \end{bmatrix}}_{[I']} \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \tag{9}$$

will result in a diagonal inertia matrix $[I']$. The inertias I_1 and I_2 of the diagonal matrix are called the principal inertias. The angle θ is defined here as the orientation of the I_1 inertia axis. Note that I_1 and I_2 are the eigenvalues of the matrix $[I]$, while the corresponding eigenvectors will determine the principal inertia axes orientations. Since $[I]$ is symmetric and positive definite, the eigenvectors are guaranteed to be orthogonal.

To compute the eigenvalues I_i of $[I]$, we need to solve the quadratic equation

$$\det\left(\begin{bmatrix} I_{xx} - I_i & I_{xy} \\ I_{xy} & I_{yy} - I_i \end{bmatrix}\right) = (I_{xx} - I_i)(I_{yy} - I_i) - I_{xy}^2 = 0 \tag{10}$$

The principal inertias I_1 and I_2 are then given by:

$$I_1 = \frac{1}{2}\left(I_{xx} + I_{yy} + \sqrt{(I_{xx} - I_{yy})^2 + 4I_{xy}^2}\right) \tag{11a}$$

$$I_2 = \frac{1}{2}\left(I_{xx} + I_{yy} - \sqrt{(I_{xx} - I_{yy})^2 + 4I_{xy}^2}\right) \tag{11b}$$

Note that here we define $I_1 \geq I_2$. Thus we can assume that I_1 will always be the largest principal inertia of the target shape.

To determine the eigenvectors of $[I]$ we could solve the standard eigenvector problem. However, since we know that for a symmetric, positive definite matrix the eigenvectors will be orthogonal unit vectors, we can take advantage of this fact by simply solving for the angle θ in Eq. (9). Carrying out the matrix multiplications in Eq. (9) leads to the four equations

$$\cos\theta I_{xx} + \sin\theta I_{xy} = I_1 \cos\theta \tag{12a}$$

$$\cos\theta I_{xy} + \sin\theta I_{yy} = I_1 \sin\theta \tag{12b}$$

$$-\sin\theta I_{xx} + \cos\theta I_{xy} = -I_2 \sin\theta \tag{12c}$$

$$-\sin\theta I_{xy} + \cos\theta I_{yy} = I_2 \cos\theta \tag{12d}$$

Multiplying Eq. (12a) by $\sin\theta$, Eq. (12b) by $\cos\theta$, and subtracting one from the other, leads to

$$\sin\theta\cos\theta(I_{xx} - I_{yy}) = (\cos^2\theta - \sin^2\theta)I_{xy} \tag{13}$$

This equation can be solved for the desired angle θ in terms of the given inertias I_{xx}, I_{yy} and I_{xy}:

$$\theta = \frac{1}{2}\tan^{-1}\left(\frac{2I_{xy}}{I_{xx} - I_{yy}}\right) \tag{14}$$

When numerically evaluating the $\tan^{-1}()$ function, it is important to use the `atan2()` function which takes both the numerator and denominator as arguments. Doing so will then return the angle θ in the proper quadrant and without singularities.

Given the principal axis orientation angle θ, it is now trivial to compute the desired eigenvectors (principal axis unit direction vectors) using:

$$\boldsymbol{v}_1 = \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix} \tag{15a}$$

$$\boldsymbol{v}_2 = \begin{pmatrix} -\sin\theta \\ \cos\theta \end{pmatrix} \tag{15b}$$

Note that the eigenvector directions are only unique to within a sign. Both θ and $\theta + \pi$ would yield the proper eigenaxis orientation angle. The snake algorithms currently keep track of the previous primary eigenaxis angle and make sure that no switching occurs between the two possible solutions.

Determining the Principal Axis Dimensions

Given the principal inertias I_1 and I_2 of the shape being tracked by the snake, it is possible to estimate some shape dimensions if we can assume that the target is of a particular shape. For example, let the target area be a rectangular shape with a half-height h and half-length l as illustrated in Figure 1. Note that we are assuming here that $l \geq h$. The principal inertias I_i are then related to the box dimensions h and l through

$$I_1 = \frac{A}{3} l^2 \tag{16}$$

$$I_2 = \frac{A}{3} h^2 \tag{17}$$

where $A = 4lh$ is the box area. These two equations can be trivially solved for the desired box dimensions:

$$l = \sqrt{\frac{3I_1}{A}} \tag{18}$$

$$h = \sqrt{\frac{3I_2}{A}} \tag{19}$$

If the true target shape is not a perfect rectangle, then this routine will approximate the equivalent box dimensions by using the principal inertias. Figure 2 illustrates the rectangular box dimensions being estimated from the snake area principal inertias. The primary inertia axis is shown in red, while the secondary inertia axis is shown in blue. The target center of mass is highlighted by a green circle. The yellow arc on the green circle illustrates the heading angle θ. Note that despite the snake rounding off the box corners, the box dimensions are estimated rather accurately. This computation is robust to small shape errors since the area integral is only minorly affected by these snake-tracking errors.

This algorithm can be modified for any general two-dimensional shapes which are defined through two parameters, as long as an analytical solution exists to extract the shape parameter from the principal inertias. If the target shape is assumed to be nearly elliptical,

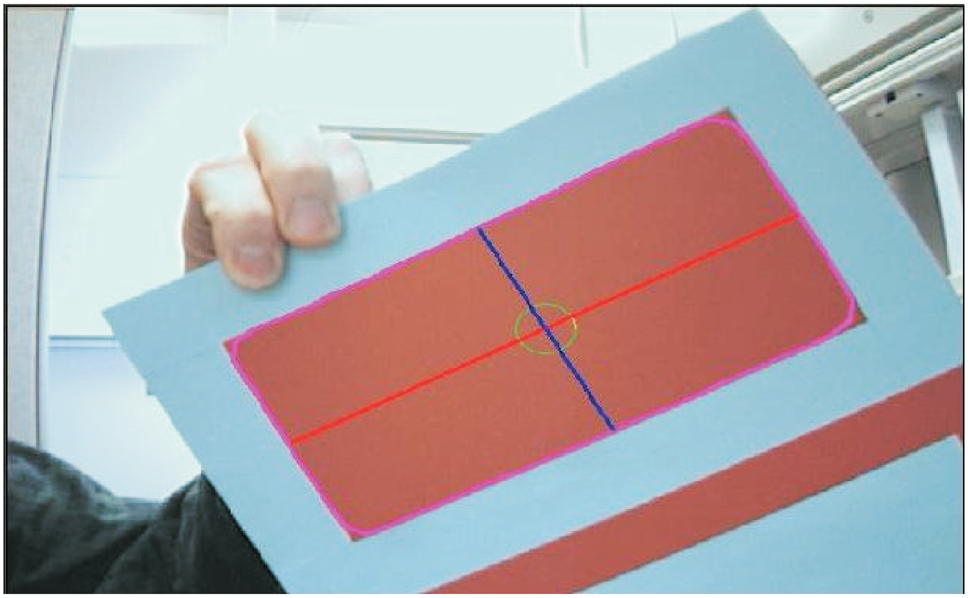

Figure 2. Illustration of the Rectangular Box Dimensions Being Estimated from the Snake Area Principal Inertias

then the semi-major axis a and semi-minor axis b are computed using

$$a = \sqrt{\frac{4I_1}{A}} \tag{20}$$

$$b = \sqrt{\frac{4I_2}{A}} \tag{21}$$

where $A = ab\pi$ is the ellipse area.

Depth Estimation Using the Principal Axis Dimensions

The (x_c, y_c) center of target coordinates computed earlier located the target in the two-dimensional image plane. Generally speaking no depth information can be extracted from a single image. However, if the target shape and dimensions are known a priori, then it is possible to use the estimated shape dimensions to extract a depth estimate. Assume a geometric feature of the target area has a known dimension d, while the projection of this feature onto the camera image plane will result in an object h pixel in size.

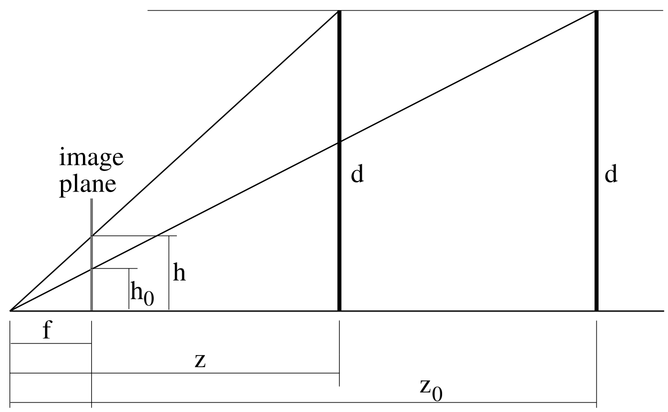

Figure 3. Illustration of Simplified Pin-Hole Camera Model Used for Depth Extraction

Figure 3 illustrates a simplified model of a pin-hole camera where f is the focal length and z is the depth of the target object. Using the geometric law of similar triangles, we find that

$$\frac{h}{f} = \frac{d}{z} \tag{22}$$

must be satisfied. If the snake algorithm has provided us with the size h of a target shape feature, then the depth of the target can be estimated using

$$z(h) = \frac{df}{h} = \frac{\gamma}{h} \tag{23}$$

where $\gamma = df$ is a constant of this specialized target tracking problem. The dimension d is assumed to be known for the target. Given the camera specifications, the parameter f might be known as well. However, to avoid dependencies on such third-party parameter information, it is simple to calibrate the camera and find the parameter γ directly. By placing the object a known distance z_0 away from the camera, and measuring the corresponding pixel dimension h_0, the calibration parameter γ is found through

$$\gamma = df = h_0 z_0 \tag{24}$$

This depth extraction algorithm assumes that the two-dimensional target shape is collinear with the camera plane. If not, then the out-of-plane rotation angle ϕ will cause some foreshortening of the measured dimension d. The apparent dimension d' is given by

$$d' = d\cos\phi \tag{25}$$

By measuring the shortened distance d' instead of d, this depth extraction algorithm will provide a depth measure which is too large by a factor of $1/\cos\phi$. Let us investigate how sensitive this depth extraction process is to out-of-plane orientations of the target shape. Using a Taylor series expansion of d', we find

$$d' \approx d - \frac{d}{2}\phi^2 + \frac{d}{4!}\phi^4 - \cdots \tag{26}$$

If the out-of-plane angle ϕ is small, then the approximation $d' \approx d$ is valid up to second order of ϕ. This means that the depth extraction algorithm is linearly insensitive to out-of-plane angles. For small out-of-plane angles, the depth measurement will only be exaggerated by a factor of $(1+\phi^2/2)$.

Conclusion

This technical report outlines how to compute target shape features using the area moments up to second order. The demanding area integrals are replaced with an algorithm that uses a line integral of the discrete snake points. The result is a very fast algorithm that is able to extract the target shape area, center of mass, orientation of the target, principal area inertias, as well as potentially the target dimensions and depth. The results are typically insensitive to small shape errors of the snake.

References

[1] Hanspeter Schaub. Statistical pressure snakes based on color images. Technical report, Sandia National Labs, Albuquerque, NM, Feb. 2003.

[2] Carsten Steger. On the calculation of moments of polygons. Technical Report FGBV–96–04, Forschungsgruppe Bildverstehen (FG BV), Informatik IX, Technische Universität München, Aug. 1996.

DISTRIBUTION:

1 Virginia Polytechnic Institute
Attn: Hanspeter Schaub
Aerospace and Ocean Engineering Department
228 Randolph Hall
Blacksburg, VA 24061-0203

1 MS 1125
Phil Bennett, 15252

1 MS 1125
Dan Marrow, 15252

1 MS 1125
Robert J. Anderson, 15252

1 MS 9018
Central Technical Files, 8940-2

2 MS 0899
Technical Library, 4916