

# MODEL BASED MULTI-VIEW ACTIVE CONTOURS FOR QUALITY INSPECTION

Pablo d'Angelo, Christian Wöhler and Lars Krüger

{pablo.d.angelo | christian.woehler | lars.krueger }@daimlerchrysler.com

*DaimlerChrysler Research and Technology*

*Machine Perception (RIC/AP)*

*P.O. Box 2360, D-89013 Ulm, Germany*

**Abstract** In this paper, 3D parametric active contours are used to segment contours of flexible objects. The contour is a 3D curve or a "weak" model supplying information about the object contour. Views from two or more viewpoints are incorporated and enable direct reconstruction in three-dimensional space, without utilising volumetric images or separate active contours on each image. This curve is optimised using a greedy algorithm, leading to a fast algorithm, suitable for use in on-line quality inspection applications. 3D ziplock ribbon snakes are used to model tubes and other approximately rotationally symmetric objects. Model information is incorporated by hard constraints and model-specific energy terms. The proposed 3D contour segmentation algorithm is applied one synthetic and two real-world scenes. The average distance between the reconstructed contour and the ground truth amounts to 1 mm, which is equivalent to approximately 1 disparity pixel for all three examples.

**Keywords:** 3D Reconstruction, Active Contour, Multi-View Reconstruction

## 1. Introduction

Segmentation of object contours is an important problem in computer vision. Many different approaches have been proposed to solve this problem, among them active contours or snakes. Since they were first introduced [Kass et al., 1988], many variations and improvements of the original snake algorithm have been proposed, such as balloon snakes [Cohen, 1991], ziplock snakes [Neuenschwander et al., 1997], gradient vector field snakes [Xu and Prince, 1998] and implicit active contour models [Caselles et al., 1995, Sethian, 1999].

We propose a parametric active contour framework to recover the 3D contours of rotationally symmetric objects, like tubes, given multiple views of a scene. It is applied to quality inspection problems. Because of their repetitive

nature, these tasks are error-prone if done by humans. Manufacturing efficiency and product quality can be improved by using robust automatic quality inspection systems. Many 3D Active contour algorithms work on volumetric data. Some algorithms [Cañero et al., 2000] implement 3D active contour curves from two images. To our knowledge, the proposed algorithm is the first 3D ribbon active contour algorithm based on multiple views.

The paper is organised as follows: First the requirements for 3D quality inspection tasks are specified. Section 2 gives a brief overview of active contour algorithms relevant to our solution. Section 3 shows how multiple views and constraints are integrated into the proposed active contour framework. Preliminary results are given in section 4. The last section provides a summary and outlines future work.

## Requirements for Active Contours in Quality Inspection Tasks

In some quality inspection tasks 3D reconstruction is needed, for example to check if a cable is mounted correctly and is not located close to dangerous parts, like hot surfaces. Other desirable properties include reliability and low computational complexity. The algorithm should be able to handle real images and be robust in the presence of partial occlusions and cluttered backgrounds. Inspection is usually an integral part of the manufacturing process and it should not increase the production time. While real-time (video frame rate) processing might not be necessary, the segmentation should be fast enough (several seconds on standard hardware) to fit into the production work flow.

Since quality inspection usually takes place in controlled environments and with known parts, some prior knowledge is available and can be used by the algorithm.

### 2. Active Contours

Active contours have been used extensively for segmentation [Hinz et al., 2001, Neuenschwander et al., 1997, Williams and Shah, 1992, Brigger et al., 2000] and tracking [Blake and Isard, 1998, Kass et al., 1988] purposes. In the Kass approaches, the basic snake is the a parametric function  $\mathbf{p}$ , (representing a contour curve or model):

$$\mathbf{p} = \mathbf{v}(s) \forall s \in [0, l] \quad (1)$$

where  $\mathbf{p}$  is a contour point at parameter  $s$ . An energy function is minimised over the contour  $\mathbf{v}(s)$ :

$$\int_0^l E_{\text{snake}}(\mathbf{v}(s)) ds \quad (2)$$

$E_{\text{snake}}$  can be separated into four terms:

$$E_{\text{snake}}(\mathbf{v}(s)) = \alpha E_{\text{cont}}(\mathbf{v}(s)) + \beta E_{\text{curv}}(\mathbf{v}(s)) + \gamma E_{\text{ext}}(\mathbf{v}(s)) + \delta E_{\text{con}}(\mathbf{v}(s)) \quad (3)$$

The internal energy  $E_{\text{int}} = \alpha E_{\text{cont}}(\mathbf{v}(s)) + \beta E_{\text{curv}}(\mathbf{v}(s))$  regularises the problem and favours a smooth contour.

$E_{\text{ext}}$ , the external energy, is based on the image at  $v(s)$  and links the contour with the image. In this paper, the gradient magnitude of the image  $I$  is used:  $E_{\text{ext}} = |\nabla \mathbf{I}(\mathbf{v}(s))|$ .

$E_{\text{con}}$  is used by the original snake [Kass et al., 1988] to introduce constraints, for example linking of an active contour point to other contours or springs. User interaction can also be cast into  $E_{\text{con}}$ . Later on, balloon snakes have emerged [Cohen, 1991], where  $E_{\text{con}}$  is used to “inflate” the active contour, to counteract the shrinking introduced by the original  $E_{\text{int}}$ .

The weight factors  $\alpha$ ,  $\beta$ ,  $\gamma$  and  $\delta$  can be chosen based on the application.

Disadvantages of the parametric model includes its dependence on parametrisation, which can lead to numerical instabilities and self intersection problems when applied to some complex segmentation tasks. Implicit active contours models avoid these problems and can handle topological changes automatically [Caselles et al., 1995]. However, they are not used in this paper due to their higher computational complexity.

A contour does not need to be a single curve and modifications to delineate ribbon structures, like roads in aerial images [Fua and Leclerc, 1990], or blood vessels in angiographic images [Hinz et al., 2001], have been proposed in the literature.

In many applications snakes are used as an interactive segmentation tool. A human operator can specify the initial contour roughly and can move the snake around possible local minima, for example with a force field attached to the mouse cursor [Kass et al., 1988].

### 3. Multiple Views and Model Information

The greedy active contours approach introduced in [Williams and Shah, 1992] is used as the basis for the 3D snake framework proposition in this paper: The contour is modelled by polyline consisting of  $n$  points and finite differences are used to approximate  $E_{\text{cont}}$  and  $E_{\text{curv}}$  at each point  $\mathbf{p}_s$ :

$$E_{\text{cont}}(v(s)) \approx ||\mathbf{p}_s - \mathbf{p}_{s-1}| - h|$$

$$E_{\text{curv}}(v(s)) \approx |\mathbf{p}_{s-1} - 2\mathbf{p}_s + \mathbf{p}_{s+1}|$$

$h$  is the mean distance between the polyline points.

The greedy minimisation algorithm is an iterative algorithm, which selects the point of minimal energy inside a local neighbourhood. The greedy optimi-

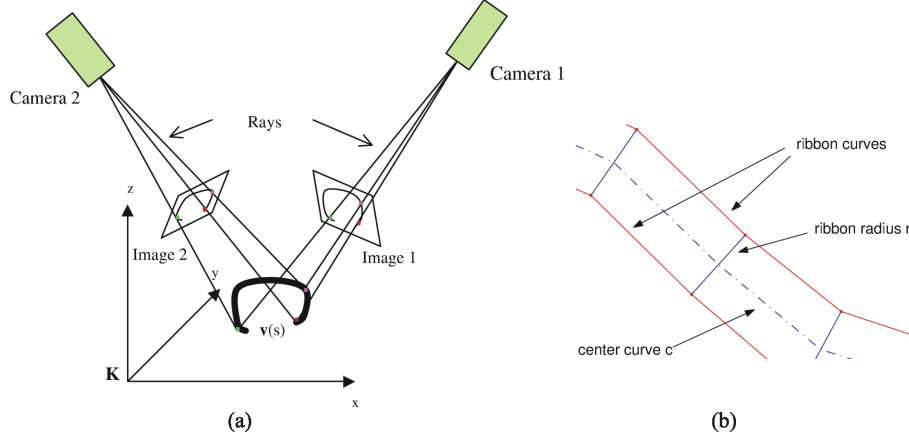


Figure 1. (a) Projection of model into multiple images 3D contour  $\mathbf{v}(s)$  curve in the world coordinate system  $\mathbf{K}$ . Two cameras are observing the scene. Calculation of the external energy requires the contour curves in the image planes of all cameras. (b) Ribbon snake.

sation is done separately for every point, from the first point at  $s = 0$  to the last point at  $s = l$ . The energy  $E(\mathbf{v}(s))$  is calculated for each candidate point  $\mathbf{p}$  in a neighbourhood grid  $\mathbf{H} \in R^d$ , where  $d$  is the dimensionality of the curve. The point of minimum energy inside  $\mathbf{H}$ ,  $\mathbf{p}_{\min}$  is selected as the new curve point at  $\mathbf{v}(s)$ . This procedure is repeated until all points have reached a stable position or a maximum number of iterations has been reached.

Since the greedy optimisation algorithm does not necessarily find a global minimum, it needs a good initialisation to segment the correct contours. Especially for the case of segmenting flexible objects, however, providing a suitable initialisation along the whole contour might not always be feasible. In such cases, the ziplock snake algorithm [Neuenschwander et al., 1997] is used. Ziplock snakes are initialised with the contour end points and their tangents. The contour is divided into active parts that use  $E_{\text{snake}}$  and inactive parts that are only influenced by the internal regularisation energy  $E_{\text{int}}$ . The force boundaries start close to the end points and travel towards each other once the boundary point has converged to a stable position.

### 3D Multi View Contours

If volumetric images are available, the image energy  $E_{\text{ext}}$  of a 3D contour can be calculated directly from the volumetric data. In industrial quality inspection applications, volumetric data is not available. But it is usually possible to obtain images acquired from multiple viewpoints. In that case  $E_{\text{ext}}$  can

be calculated by projecting the contour onto the image planes of the cameras, see Fig. 1(a).

We use the camera model described in [Bouguet, 1997]. The intrinsic and extrinsic parameters are obtained by multiocular camera calibration [Krüger et al., 2004].

Any number of images,  $i = 1..n$ , can be used for this projection. The camera matrix  $K_i$  of each camera is known and contains both internal and external camera parameters. It is used to project a point in world coordinates  $p_w$  onto the corresponding image point:  $p_{img,i} = T_i K_i p_w$ .

The energy term  $E_{ext}$  that connects the contour with the images is calculated based on the projection of the contour into each image plane according to  $E_{ext}^* = \sum_{i=1}^n E_{ext}(p_{img,i})$ , where  $p_{img,i}$  are the projection of the contour onto the images.

Creation of a separate active contour for each image or a depth map created by traditional stereo is therefore not necessary. All image and constraint information is used by the energy minimisation step, in the model parameter space.

In our examples, we use the image gradient,  $E_{ext}(\mathbf{p}_{img}) = |\nabla \mathbf{I}(\mathbf{p}_{img})|$ , as external energy term.

Occlusions of the object are not considered during modelling, but the algorithm can handle partial occlusions and holes in the contour if they do not appear excessively.  $E_{int}$  and  $E_{con}$  are independent of the occlusions happening in different views, and provides information in these cases.

The described approach requires that either the points of the extracted 3D snake correspond to the same scene points, respectively, or that the object displays certain symmetries. In our scenario of inspection of tubes and cables, the objects to be segmented are rotationally symmetric with respect to a centerline, such that their silhouette is similar from multiple viewpoints and can be described by a center curve  $(x(s), y(s), z(s))^T$  and a radius  $r(s)$ , leading to  $\mathbf{v}(s) = (x, y, z, r)^T$ .

In this case, the contour model is not a simple curve and can not be projected into the images as described above. The center line  $(x, y, z)^T$  is projected into the images, and the corresponding ribbon curves are calculated by projecting the radius  $r$  into the image planes.

## Model Information

If prior knowledge about the contour is known, it can be integrated into the snake optimisation algorithm to improve its convergence. In a manufacturing environment, CAD models of the parts that are subject to quality inspection are available and can be used to provide prior knowledge to the active contour segmentation. If the shape of objects should be recovered, only constraints

that are invariant to the possible shapes of the object should be used. For quality inspection tubes, this includes elasticity, length, radius and sometimes the mounting position, depending on the application.

The model information is introduced into the optimisation process in two ways. The first is by adding additional energy terms, the second by using a constrained optimisation algorithm. Additional model-based energy terms that favour a certain shape of the contour can be added to  $E_{\text{con}}$ .

For example, the approximate radius of a cable is known, but it can vary at some places due to labels or constructive changes. A 3D ribbon snake is used to detect the cable shape and position, then  $E_{\text{con}} = E_{\text{rib}} = (r(s) - r_{\text{model}}(s))^2$  can be used as a “spring energy” to favour contours with a radius  $r$  close to a model radius given by the function  $r_{\text{model}}(s)$ . However, adding constraint terms to the objective function may result in an ill-posed problem with poor convergence properties and adds more weight factors that need to be tuned for a given scenario [Fua and Brechbühler, 1996].

The second approach is to enforce model constraints through optimiser constraints. In the greedy algorithm this is done by intersecting the parameter search region  $H$  and the region permitted by the constraints  $C$  to obtain the allowed search region  $H_c = H \cap C$ . This ensures that these constraints can not be violated. They are also called hard constraints.

For some applications like glue line detection, the surface in which the contour is located is known, for example from CAD data. In other cases, the bounding box of the object can be given. These facts can be exploited by a constraint that restricts the optimisation to the corresponding surface or volume.

Model information can also be used to create suitable initial contours. For example tubes are often fixed to other parts with brackets. The pose of these brackets can be given a priori, when repeated quality inspection tasks are performed, or determined by using pose estimation algorithms [Gavrila and Philomin, 1999]. These points can be used as start and end points (boundary conditions) for 3D ziplock ribbon snakes.

## 4. Results

The algorithm has been tested on both artificial and real scenes. For all examples, a 3D ribbon snake was used. The weight factors of Eq. (3) were set to  $\alpha = 1$ ,  $\beta = 1$ ,  $\gamma = 3$  and  $\delta = 0$ . Additionally a hard constraint has been placed on the minimum and maximum ribbon width. This avoids solutions with negative or unrealistic large width.

To estimate the quality of fit, an artificially rendered scene was used as a test case. Figure 2 (a) and (b) show the example scene and its 3D reconstruction result. The start and end points and their tangents have been specified. In a

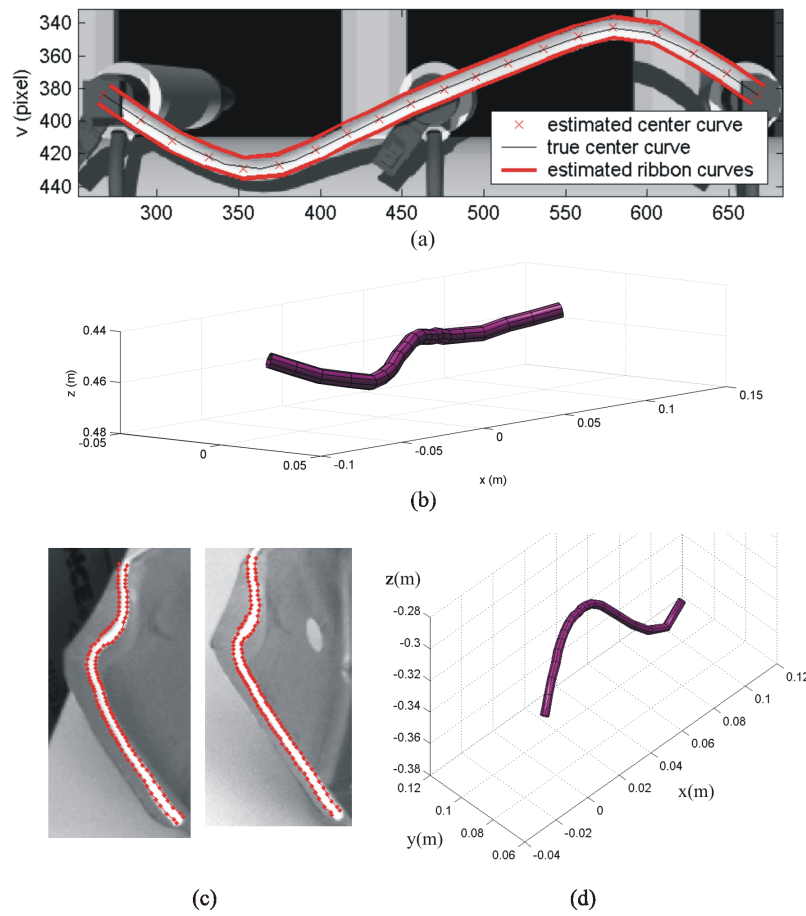


Figure 2. (a) One of the three artificially rendered input images with the overlaid reprojections of the 3D ribbon snake. A virtual trinocular camera with a resolution of 1024x768 pixels and a base distance of 10 cm has been used to create the images. (b) Reconstructed tube, shown from a different viewpoint. The mean distance between ground truth and reconstruction is 1.5 mm. (c) Real-world images showing a glue line on a non-planar surface. (d) Resulting 3D ziplock ribbon snake.

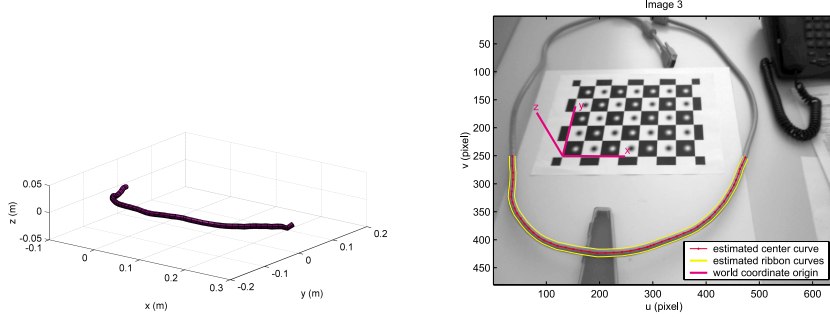


Figure 3. Monitor Cable Scene. It can be seen that the reconstructed cable is located above the table plane ( $z=0$ ). The world coordinate system has been defined by the pose estimation of the chessboard.

real application these could either be taken from a CAD model, or estimated by pose estimation of the anchoring brackets.

The ground truth used to produce the image is known and is compared to the segmented contour. The metric used is the average distance of the ribbon snake's centre curve to the model centre curve.

For similarly parametrised, discrete polylines, the following formula has been used:

$$d = \frac{\sum_{i=1}^{nP-1} \frac{A_i}{l_i}}{nP} \quad (4)$$

where  $P_i, i \in [1, nP]$  are the polygon points and  $A_j, j \in [1, nP - 1]$  is the area between two corresponding polygon spans of  $P_m$ , the model polygon and  $P_c$ , the contour polygon.  $A_j$  is calculated using  $A_j = |\text{area}(P_{c_j}, P_{c_{j+1}}, P_{m_j})| + |\text{area}(P_{m_j}, P_{m_{j+1}}, P_{m_{j+1}})|$ .

The average distance between the ground truth and the segmented curve is for this model is 1.5 mm, which roughly corresponds to 1 pixel disparity error. Subpixel greedy stepwidths and interpolated image energy calculation were used to obtain this result.

Furthermore, the algorithm was applied to two different real-world scenes acquired with a trinocular camera (Figure 2c-d, Figure 3). A 3D ziplock ribbon snake has been used in these examples. The approximate ribbon radius was used as model information, and constraints for the upper and lower bound of the ribbon width were used. The segmented contour part is displayed from a slightly different angle to show the spatial segmentation. The accuracy of reconstruction compared to ground truth obtained with a calliper gauge is better than 1 mm in both example scenes.

The behaviour of the algorithm in the presence of partial occlusions has been tested by moving a virtual rod over the scene, see Figure 4 and Table 2.

Test case	Mean error (mm)	Relative error (pixel)
Synthetic	1.5	1
Glueline	$\sim 1$	1.6
Monitor cable	$\sim 1$	1.5

Table 1. Reconstruction errors

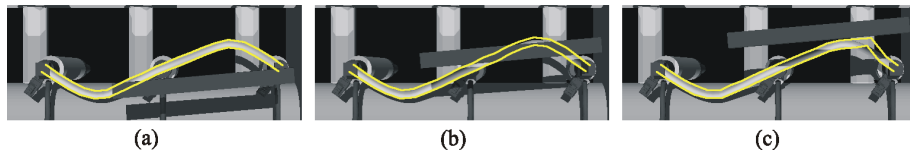


Figure 4. Partial occlusion. A virtual rod has been moved over the example of Figure 2 (a,b).

Test case	Mean error (mm)	Runtime (s)
(a)	1.1	7
(b)	3.1	11
(c)	3.8	13

Table 2. Reconstruction error under partial occlusion, see Fig. 4. The time required to fit the contour increased with the reconstruction error.

All test cases have been run on a Pentium Mobile with 1.7 GHz. The fitting took between 1 and 23 seconds, depending on the complexity of the scene and the optimiser parameters.

## 5. Summary and Conclusion

We have proposed an active contour approach for model based segmentation of 3D contours. Multiple views of the object are used during the segmentation process. Model information that is invariant with respect to the possible defects of the parts is incorporated into model specific energy terms or minimisation constraints and improves the segmentation performance.

The proposed algorithm is able to perform an accurate and robust 3D contour segmentation in real world scenes. It has been applied to the scenario of 3D shape reconstruction for quality inspection applications. The stability of the algorithm is mainly due to the usage of model based constraints and initial contours curves based on model information. It is, however sensitive to occlusions and self-intersections if they appear excessively. The current approach is limited to lines and tube shaped objects, which is sufficient for many applications. If other shapes should be supported, projection and optimisation needs to be adjusted.

Future work includes improvements of the optimisation algorithm, the investigation of fast implicit contour models to improve the performance with complicated scenes and avoid parameterisation problems. Integration of active contour algorithms and object pose estimation into a combined system is also an interesting topic.

## References

- [Blake and Isard, 1998] Blake, A. and Isard, M. (1998). *Active Contours*. Springer-Verlag, London.
- [Bouguet, 1997] Bouguet, J. (1997). Camera calibration toolbox for MATLAB. [www.vision.caltech.edu/bouguetj/calib.doc](http://www.vision.caltech.edu/bouguetj/calib.doc).
- [Brigger et al., 2000] Brigger, P., Hoeg, J., and Unser, M. (2000). B-Spline Snakes: A Flexible Tool for Parametric Contour Detection. *IEEE Transactions on Image Processing*, 9(9):1484–1496.
- [Caselles et al., 1995] Caselles, V., Kimmel, R., and Sapiro, G. (1995). Geodesic active contours. In *ICCV*, pages 694–699.
- [Cañero et al., 2000] Cañero, C., F., V., Mauri, J., and Radeva, P. (2000). Predictive (un)distortion model and 3d reconstruction by biplane snakes. In *Proceedings of International Conference on Pattern Recognition*.
- [Cohen, 1991] Cohen, L. D. (1991). On active contour models and balloons. *Computer Vision, Graphics, and Image Processing. Image Understanding*, 53(2):211–218.
- [Fua and Brechbühler, 1996] Fua, P. and Brechbühler, C. (1996). Imposing Hard Constraints on Soft Snakes. In *ECCV (2)*, pages 495–506.
- [Fua and Leclerc, 1990] Fua, P. and Leclerc, Y. G. (1990). Model driven edge detection. *Machine Vision and Applications*, 3(1):45–56.
- [Gavrila and Philomin, 1999] Gavrila, D. M. and Philomin, V. (1999). Real-time object detection for smart vehicles. In *ICCV99*, pages 87–93.
- [Hinz et al., 2001] Hinz, M., Toennies, K. D., Grohmann, M., and Pohle, R. (2001). Active double-contour for segmentation of vessels in digital subtraction angiography. In *Proceedings of SPIE*, volume 4322, pages 1554–1562.
- [Kass et al., 1988] Kass, M., Witkin, A., and Terzopoulos, D. (1988). Snakes: Active Contour Models. *International Journal of Computer Vision*, 1(4):321–331.
- [Krüger et al., 2004] Krüger, L., Wöhler, C., Würz-Wessel, A., and Stein, F. (2004). In-factory calibration of multiocular camera systems. In *Photonics Europe*.
- [Neuenschwander et al., 1997] Neuenschwander, W., Fua, P., Iverson, L., Szekely, G., and Kubler, O. (1997). Ziplock Snakes. In *International Journal of Computer Vision*, 25(3):191–201.
- [Sethian, 1999] Sethian, J. (1999). *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science*. Cambridge University Press.
- [Williams and Shah, 1992] Williams, D. J. and Shah, M. (1992). A Fast Algorithm for Active Contours and Curvature Estimation. *Image Understanding*, 55(1):14–26.
- [Xu and Prince, 1998] Xu, C. and Prince, J. L. (1998). Snakes, shapes, and gradient vector flow. *IEEE Transactions on Image Processing*, 7(3):359–369.