Shape regularized active contour based on dynamic programming for anatomical structure segmentation

Tianli Yu[1], Jiebo Luo[*2], Amit Singhal[2], Narendra Ahuja[1]

[1]Department of Electrical and Computer Engineering, University of Illinois, Urbana-Champaign, IL, USA 61801
[2]Research and Development Laboratories, Eastman Kodak Company, Rochester, NY, USA 14650

ABSTRACT

We present a method to incorporate nonlinear shape prior constraints into segmenting different anatomical structures in medical images. Kernel space density estimation (KSDE) is used to derive the nonlinear shape statistics and enable building a single model for a class of objects with nonlinearly varying shapes. The object contour is coerced by image-based energy into the correct shape sub-distribution (e.g., left or right lung), *without* the need for model selection. In contrast to an earlier algorithm that uses a local gradient-descent search (susceptible to local minima), we propose an algorithm that iterates between dynamic programming (DP) and shape regularization.

DP is capable of finding an optimal contour in the search space that maximizes a cost function related to the difference between the interior and exterior of the object. To enforce the nonlinear shape prior, we propose two shape regularization methods, global and local regularization. Global regularization is applied after each DP search to move the entire shape vector in the shape space in a gradient descent fashion to the position of probable shapes learned from training. The regularized shape is used as the starting shape for the next iteration. Local regularization is accomplished through modifying the search space of the DP. The modified search space only allows a certain amount of deformation of the local shape from the starting shape. Both regularization methods ensure the consistency between the resulted shape with the training shapes, while still preserving DP's ability to search over a large range and avoid local minima.

Our algorithm was applied to two different segmentation tasks for radiographic images: lung field and clavicle segmentation. Both applications have shown that our method is effective and versatile in segmenting various anatomical structures under prior shape constraints; and it is robust to noise and local minima caused by clutter (e.g., blood vessels) and other similar structures (e.g., ribs). We believe that the proposed algorithm represents a major step in the paradigm shift to object segmentation under nonlinear shape constraints.

Keywords: nonlinear shape prior constraints, kernel space density estimation, active contour, dynamic programming, shape regularization, anatomical structure segmentation.

1. INTRODUCTION

Segmentation of anatomical structures is often a critical component of a medical imaging system, such as computer-aided diagnosis (CAD) and patient information system (PIS). In chest radiography, which accounts for 70% of all x-rays, researchers have developed numerous methods for segmenting lung fields, rib cage, heart, clavicle, blood vessels, as well as abnormal structures, such as lung nodules. However, given the projection nature of chest radiography, superimposed anatomical structures make image complicated. Combined with imaging noise, chest radiography poses a great challenge to diagnosis; even experienced radiologists sometimes have trouble discerning infiltrates from the normal patterns of blood vessel branches in the lung fields, or detecting subtle nodules that indicate cancer.[1] In addition, significant inter-observer, and even intra-observer, differences occur when radiologists outline the boundaries of normal structures or rate the severity of abnormal findings. The clinical importance of

[*] {jiebo.luo, amit.singhal}@kodak.com
[1] tyu@uiuc.edu; ahuja@vision.ai.uiuc.edu

chest radiography and its complicated nature has driven continued research activities in developing robust segmentation algorithms to assist radiologists and improve automation.

While the nature of chest radiography gives rise to all the challenges for a robust algorithm, the limited domain of the problem provides an opportunity to incorporate prior knowledge of the shape, and in some cases the appearance, of the anatomical structures of interest. After the early years, which focused on pixel-classification and rule-based reasoning, parametric shape models (ellipses, parabolas, polynomials, etc.) were first introduced to segment structures such as ribs.[2] However, the over-simplification associated with such models frequently limits the accuracy of the segmentation results. A more recent and increasingly popular approach is to learn the shape (and appearance, when applicable) of an object *statistically* from a set of training examples, and then constrain the segment contour to a sub-manifold of plausible shapes during the segmentation process. For specific, known structures, this approach has been shown to significantly improve segmentation results, the most notable being the active shape model (ASM) used for lung field segmentation.[3]

Although the shape prior can be effective in counteracting clutter, occlusion, and noise in images, existing approaches, such as the ASM, are limited in their ability to handle more complicated shape variations for many anatomical structures. In essence, the permissible shapes are assumed to form a multivariate Gaussian distribution, and thus, all allowable shape deformations have to be linear combinations of a set of eigen shapes, such as those given by principal component analysis (PCA).[3] It also means that any linear morphing between two permissible shapes is permissible as well. These assumptions and implications can be violated in many applications. For example, the average shapes of the left and right lung fields do not exist, and furthermore, the respective shapes of each lung field, themselves, are not tight Gaussian distributions. Consequently, with a Gaussian assumption (as imposed by PCA in ASM), the space of acceptable shapes would be far too loose for the shape prior to enforce the correct amount of constraint to the shape deformations. Such ineffectiveness of the shape model may be the reason why researchers resorted to customized features,[3] or even semi-automatic means[4] to improve the accuracy of the ASM models. In addition, the strong appearance component within the ASM may turn out to be a liability to some extent because the appearance of many anatomical structures can vary dramatically. For example, the boundary of the lung field appears different from patient to patient, from exposure to exposure, and even from segment to segment, depending on whether it is between the lung field and heart or between the lung field and diaphragm.

More recently, a new class of shape models, different from the ACM, emerged based on the framework of active contours[5] and nonlinear shape models.[1] Active contours attracted early attention in the 1990s as a way to impose shape constraints into contour-based segmentation.[3] However, the shape constraints in classic active contours are of general nature, such as smoothness, minimum length, etc. Active contours started evolving in the direction of object-specific shape constraints. First, linear shape statistics were introduced into the active contour model in a Mumford-Shah formulation, resulting in the so-called *diffusion snake*.[5] About the same time, nonlinear density estimate was first introduced in Ref. 1. These two pieces of work eventually culminated into the variational segmentation algorithm that utilizes shape statistics derived from kernel space density estimation. [7]

It is desirable to develop a method to incorporate nonlinear shape prior constraints into segmenting different anatomical structures (e.g., lung fields, clavicles, ribs) in medical images. Moreover, this method should serve as a flexible segmentation tool for a class of objects with nonlinearly varying shapes without the need to build separate models for sub-classes of the same structure. In Fig. 1, a desirable algorithm for lung field segmentation should start from the *same* elliptical shape and automatically converge to the correct shape without the need of model selection (in other words, without external knowledge whether it is the left or right lung field). In addition, the method should also be robust to noise and clutter, which are commonly seen in medical images.

(a) (b) (c) (d)

Figure 1: Lung field segmentation examples. Starting from the *same* elliptic shape, a desirable algorithm can automatically converge to the correct shape without the need of model selection (whether it is the left or right lung field). (a) Initial contour of the right lung, (b) final contour of the right lung, (c) initial contour of the left lung (note that it is the *same* as that of the right lung), and (d) final contour of the left lung.

2. METHODOLOGY

2.1 Shape regularized active contour

Our segmentation method is called shape regularized active contour, or ShRAC. It uses kernel space density estimation[2] to learn the nonlinear shape prior, which is used to regularize a dynamic programming-based active contour algorithm. ShRAC consists of two stages, the shape learning stage and the segmentation stage. It can be summarized by the diagram in Fig. 2.

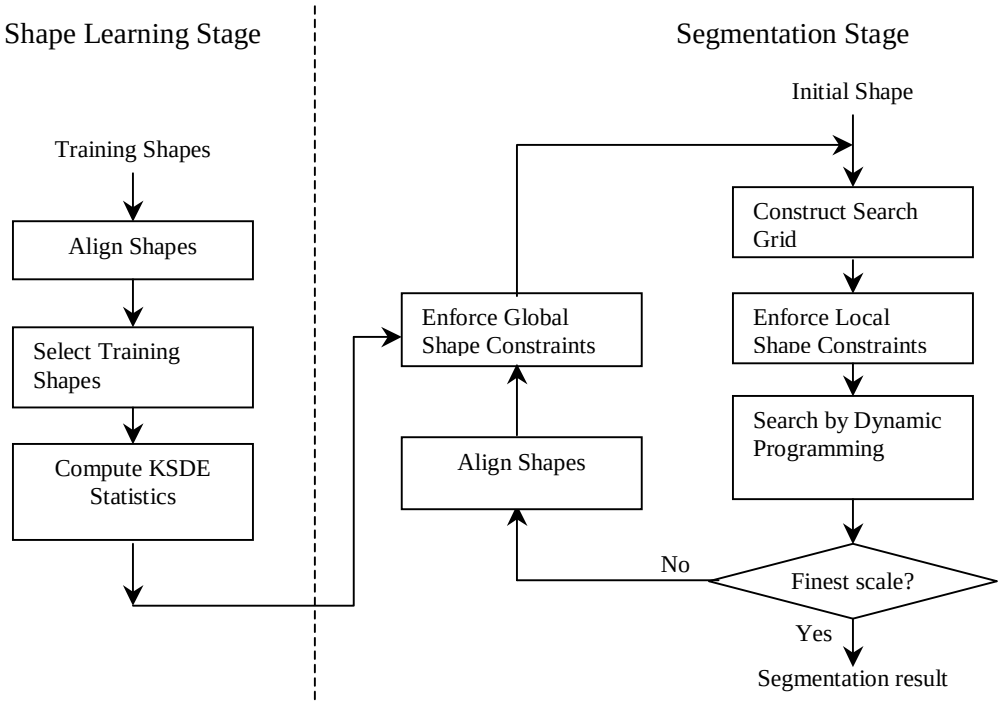


Figure 2: Block diagram of the shape regularized active contour (ShRAC) algorithm.

2.2 Shape learning

Shape learning consists of collecting shape statistics from training shapes and estimating the distribution of shapes in the shape space. We adopted the method proposed by Cremers et al.[2] to derive a nonlinear density estimation of shape prior. According to Ref. 2, the estimated density function for a shape vector *Z* can be written in the shape energy form as:

$$E_\phi(z) = \sum_{k=1}^{r}\left(\sum_{i=1}^{m}\alpha_i^k \tilde{k}(z_i, z)\right)^2 \cdot (\lambda_k^{-1} - \lambda_\perp^{-1}) + \lambda_\perp^{-1} \cdot \tilde{k}(z, z) \tag{1}$$

where z is an input shape vector, z_i is the *i*-th training shape, $\{a_i^k\}$ is the *k*-th eigenvector of the center kernel matrix, and λ_k is the corresponding eigenvalue. In addition, $\tilde{k}(\bullet,\bullet)$ is the centered kernel function of two shape vectors, and $\lambda_\perp$ is the replacement for all zero and small eigenvalues. In our implementation, we set $\lambda_\perp$ to the average of the eigenvalues that are below a threshold. A lower energy in (1) means the shape is more likely to be generated by the same source as the training shapes.

The kernel function we use in (1) is the Gaussian kernel.

$$k(x, y) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(-\frac{\|x - y\|^2}{2\sigma^2}\right) \tag{2}$$

In order to represent the shape in a vector and remove the possible variations due to rigid transformation and different point correspondences, we apply a shape alignment process to each contour and transform it into a set of control points on the contour. First, each contour is discretized into a set of contour points. These points are used to compute the major axis and standard deviation of the shape in the image plane. Next, the contour is rotated to align the major axis to the x-axis and is scaled using the corresponding standard deviation. A set of control points is selected on the contour and their coordinates form the shape vector. The control-point selection process should be consistent in roughly selecting the corresponding points on different shapes. In our implementation, we place contour points in an equally spaced fashion, starting from the two tips of the shape on the major axis.

The complexity of evaluating $E_\phi(z)$ in (1) depends on the number of training examples. $E_\phi(z)$ will be evaluated many times during the segmentation stage, so we prefer the number of training examples to be small. To this end, we employ a shape selection process before computing the shape statistics. It only selects a training shape if its distance in the shape space, compared to all other already selected shapes, is greater than a certain threshold. In that way, the shape-selection process prunes the training shapes to a manageable size and allows the computation to be focused on shapes with meaningful, significant variations.

Fig. 3 shows a set of selected training shapes for *both* left and right lungs (after alignment by the tips of the lung fields). Standard PCA, as well as KSDE, are used to estimate the probability density function, and the resulting shape energy function, along the two eigenvectors that correspond to the largest two eigenvalues from PCA, are plotted in the bottom of Fig. 3. We can see clearly that density estimation from KDSE shows two distinct clusters in the shape space, which is a better characterization of the training shape distribution (with two distinctive clusters for the left and right lung fields, respectively).

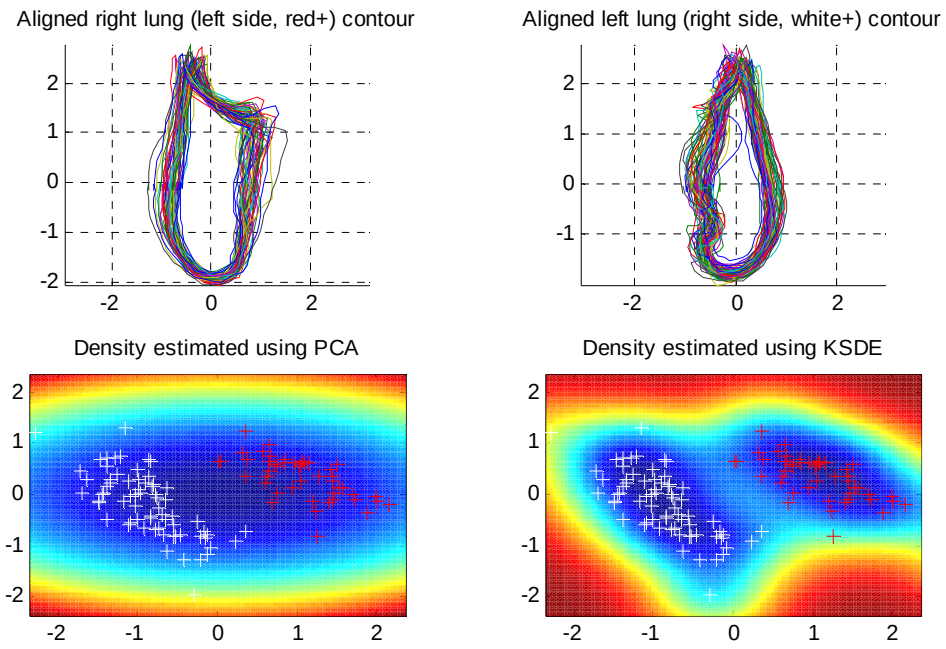


Figure 3: Comparisons of PCA and KSDE from the training data.

2.3 Active contour segmentation based on dynamic programming

We use a classic active contour segmentation formulation. We define an object boundary as a closed contour c that minimizes the average intensity difference between the interior and exterior of the object. If $h(\vec{p})$ is a function that measures the interior/exterior intensity difference on the boundary point $\vec{p}$, the cost function to be minimized is written as:

$$f(c) = \oint_{\vec{p}\ on\ c} h[\vec{p}(s)]ds \Big/ L(c) \tag{3}$$

where $L(c)$ is the length of the contour, and $\vec{p}(s)$ is the parameterized representation of the contour.

In order to use a combinatorial search, we discretize the contour into a set of control points and rewrite the cost function as the sum of the cost over line segments that link between consecutive control points.

$$f_d(c) = \sum_{i=1}^{n} h_d[l_i] \cdot |l_i| \Big/ \sum_{i=1}^{n} |l_i| \tag{4}$$

where $h_d[l_i]$ is the discretized version of $h(\vec{p})$ that computes the intensity difference on a line segment. In (4), the segment length term $|l_i|$ is used to keep the intensity differences weighted evenly along the contour. If we can ensure that the line segments are always of similar length, as we will always do in the shape alignment process, we can drop the length term in (4) and use (5), instead, as the cost function:

$$f_d(c) = \sum_{i=1}^{n} h_d[l_i] \tag{5}$$

The cost function in (5) is written as a sum of functions of segments connecting consecutive control points. Its search space, consisting of the coordinates of all control points, can be discretized and optimized efficiently using

dynamic programming (DP).[3] Note that this is in contrast to the original algorithm by Cremers et al.,[2] which uses a local gradient-descent search. The main motivation is that local gradient descent is susceptible to local minima, while dynamic programming is a good way to avoid local minima.
The proposed search can be done as follows:

Starting from an initial contour, determine the search direction for each of the control points. The search direction can be the normal direction along the contour, the radial direction from the object center, or other suitable direction, depending on the object shape and the desired search behavior.

Construct a search grid by placing a set of points at a fixed step size along each search direction, whereas these points are the candidate positions of the final control points (Fig. 4.).

Use dynamic programming to find a set of control points that minimize the cost function in (5).

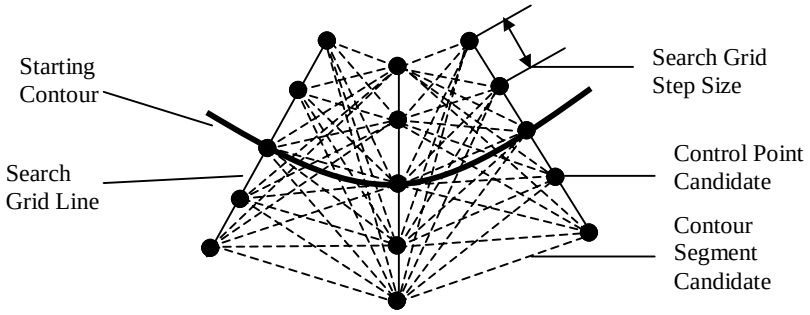


Figure 4: Constructed search space and possible line segments for an initial contour.

The advantage of DP over other continuous-domain search methods is that it finds the global minimum in the search space; therefore, it is very robust against inaccurate initialization and noise. The search space can be made quite large without worrying about being trapped in local minima. Unlike the continuous-domain search method, no gradient information is needed. Dynamic programming only uses the cost function value in (5) to perform the search. The overall complexity of DP is $O(m^2 n)$, where *m* is the number of candidate positions a control point can take in the search space, and *n* is the number of control points on the contour.

In our implementation, we use the signed intensity difference

$$h_d(l) = a_1 I_{in}(l) - a_2 I_{out}(l) \tag{6}$$

in the cost function (5), where a_1 and a_2 are the relative weighting factors between interior and exterior intensities. I_{in} and I_{out} are the average intensities inside a window on the interior and exterior side of the line segment, respectively. To make the algorithm more efficient, we apply DP in a multi-scale fashion, with the search step size and the windows size used in computing the $h_d(l)$ decrease at every iteration.

2.4 Shape regularization

Dynamic programming by itself does not take into account any shape knowledge. In fact, in order to integrate the shape energy as part of the cost function into the framework of dynamic programming, the energy function must be decomposed into a sum of functions over contour segments. However, the shape energy function from (1) contains the inter-relations of all of the control points, and therefore cannot be decomposed easily. Previous work in Ref. 2 chose to add the shape energy into the image cost function and tried to optimize it via a gradient-descent search. This approach is susceptible to local minima. In this work, we propose two methods to regularize the result from dynamic programming, based on the shape energy function in (1). Combined with the global optimality of DP, our method shows high robustness against local minima while still being able to enforce the shape prior constraints.

The first method is *global regularization*, which is similar to that used by the active shape model (ASM).[4] In ASM, after the search of landmark points, the entire shape vector is projected to the boundary of the hyper-ellipse that represents the region of most probable shapes. We take a similar approach in this work. After the dynamic programming finds the optimal contour, we compute the shape energy of the contour. If the shape energy is higher than a preset threshold, which means it is outside the distribution of probable shapes, we modify the control point vector to move it back to the realm of probable shapes. This is done through steepest gradient descent to decrease the shape energy function until it reaches below the threshold.

The disadvantage of the global regularization is that it can only be applied after the dynamic programming search finishes. This means that there is little control over whether the regularized shape would still be aligned well to the intensity boundaries in the image. It is desirable that shape regularization be enforced during the dynamic programming-based search. Therefore, we further propose to add local regularization to enforce shape priors through directly modifying the search space of DP.

Because of the use of a large search range, the resulting shape from dynamic programming may be quite different from the initial shape. In Fig. 5(a), we can see that the search space contains segments with wide varieties of lengths and orientations. Although the global regularization method will regularize the shape into an acceptable shape after each iteration, it is still possible for dynamic programming to move to a very different shape in the next iteration. Such oscillations were observed during our initial experiments. Although these oscillations can be reduced by limiting the search range in the next iteration, this will also reduce the optimality of the solution. Ideally, we prefer the next dynamic programming search to be limited to those similar to the regularized shape. In light of this consideration, we modify search space by pruning those segments that deviates far away from the initial shape from the search space.

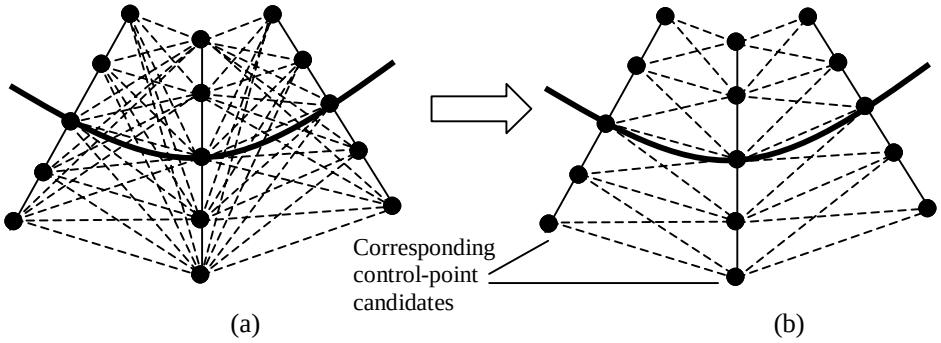


Figure 5: (a) Initial search space for DP. (b) search space after the local regularization (bandwidth = 3).

After constructing the search space by placing equally spaced control points along the contour, we only preserve segments that connect corresponding control-point candidates and their immediate neighbors, and we remove the remaining connections. In the original dynamic programming scheme, the connection matrix between the two sets of control-point candidates is a full matrix. However, in the modified search space, the connection matrix becomes a sparse, banded matrix. We define the bandwidth as the number of nearest neighboring candidate points in other sets to which a point can connect. Varying the bandwidth can effectively control the amount of local deformation allowed in the search space.

Local regularization is very effective in limiting the search space to similar shapes. Oscillations can be largely removed without severely limiting the search range. Furthermore, local regularization also helps reduce the amount of computation for the DP search from $O(m^2 n)$ to $O(mn)$.

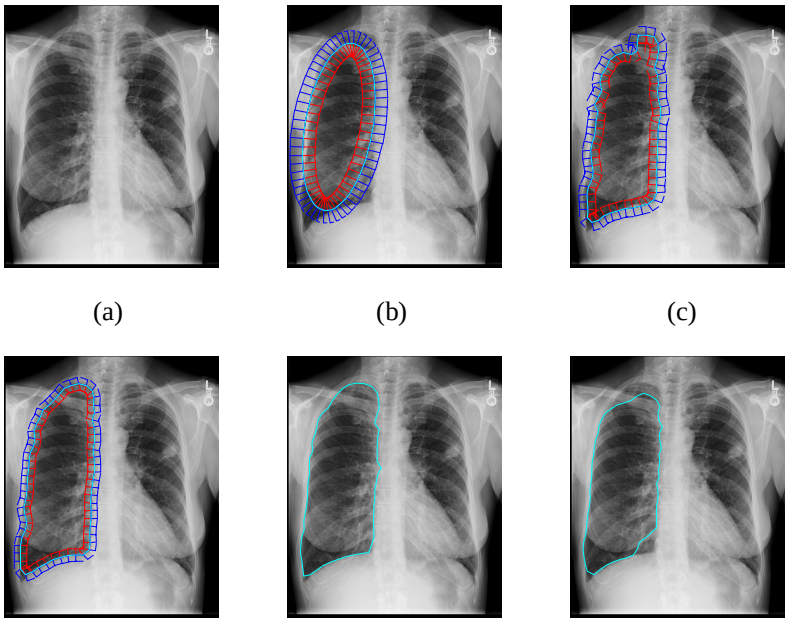
3. EXPERIMENTAL RESULTS

We performed tests of the proposed ShRAC algorithm on two segmentation tasks for radiographic images: the lung field segmentation and clavicle area segmentation. Both experiments show that ShRAC is effective in segmenting a wide variety of objects from cluttered background, according to learned shape prior.

3.1 Lung field segmentation

ShRAC is designed to be flexible with respect to nonlinear shape variations so that we only need to train one shape model for both left and right lung fields. We have two data sets (a total of 383 images, 766 shapes for both lung fields) that are manually outlined as the ground truth. We tested ShRAC on each dataset using the other as the training set. For each training set, the automatic shape-selection process selects about 100 shapes, as training examples, to compute the KSDE statistics. Our shape model is represented as a single, closed contour using 60 control points, regardless if it is the left or right lung fields.

The segmentation process is illustrated in Fig. 6. We initialize ShRAC using an algorithm that extracts a rough centerline of the lung field. The position, orientation, and scale of the centerline were used to construct an ellipse used as the starting shape (Fig. 6(b)). Fig. 6(b) also shows the initial windows to compute interior/exterior intensity difference in red/blue lines. The initial window width in the contour normal direction is 30 pixels, and the search grid spans 72 pixels on both sides of the contour. The initial search grid is made wide enough to compensate for possible poor initialization. The window width, as well as the search grid, is reduced gradually at each iteration to refine the shape. During the actual segmentation, we first perform two iterations of the algorithm without shape regularization to allow the contour to roughly expand to the correct size. The result of these two iterations is shown in Fig. 6(c). Because the bones (clavicle, ribs) show in the image as a high-intensity area, the contour was initially attracted to these structures. This is because dynamic programming tries to find a global minimum in the area covered by the search grids. Next, the shape regularization is enforced and the resulting shape is shown in Fig. 6(d). Fig. 6(e) shows the final segmentation result.



(a) (b) (c)

(d) (e) (f)

Figure 6: Segmentation process of lung field: (a) Input image, (b) initial contour, (c) contour after first two dynamic programming search, (d) contour after shape regularization, (e) final contour produced by ShRAC, and (f) final contour from QN optimization.

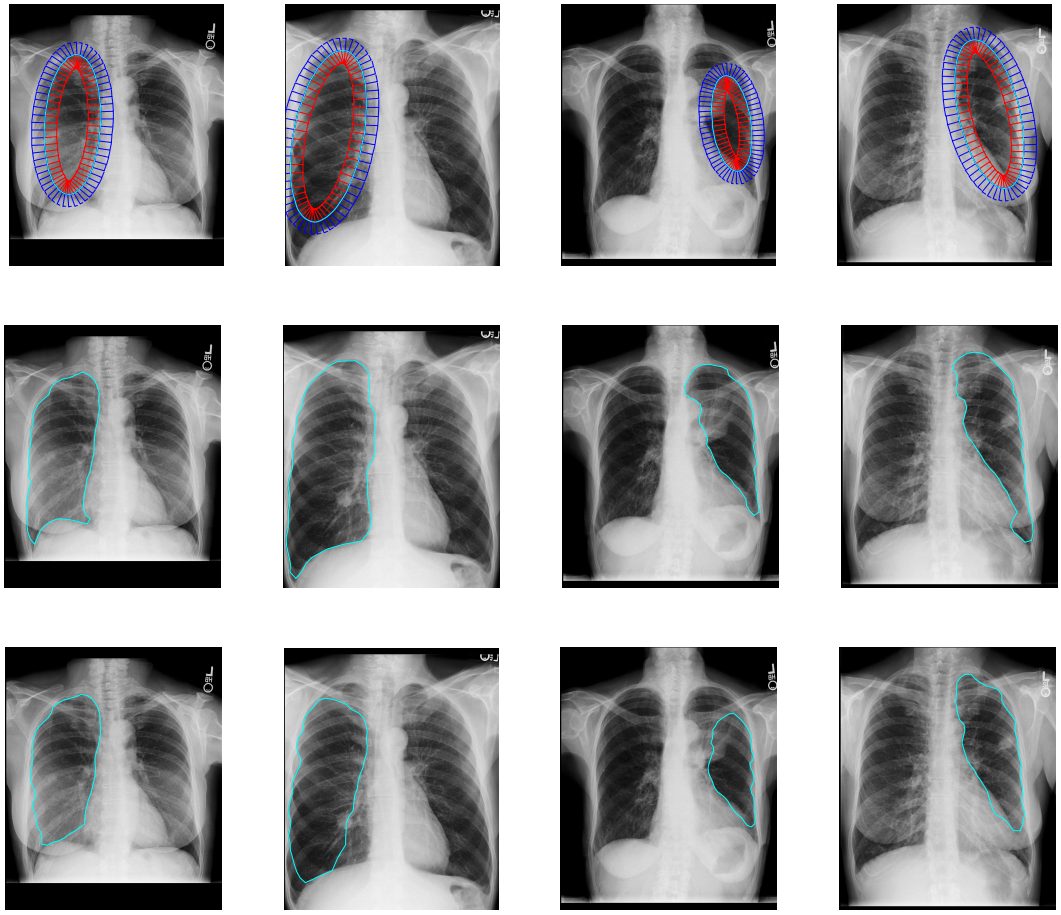


Figure 7: Segmentation results by ShRAC and QN optimization, respectively: First row: initial contours. Second row: final contours from ShRAC. Third row: final contours from QN optimization algorithm.

For comparison purposes, we implement an alternative algorithm based on the BFGS Quasi-Newton method[5] (referred heretofore as QN optimization) that minimizes a weighted sum of the intensity difference term and the shape energy term using local gradient information (similar to the one proposed in Ref. 2). The result from QN optimization is shown in Fig. 6(f). Clearly, a local optimization technique is prone to be caught in local minima (caused by clavicles, ribs, blood vessels, and other structures in the image), and it only produces a good result when initialization is close enough. More segmentation examples by ShRAC can be found in Fig. 7, demonstrating the efficacy of ShRAC.

For the two data sets we have, we compare the segmentation accuracy between the two algorithms in Table 1. The accuracy is computed using the overlap measure Ω, as proposed in Ref. 4:

$$\Omega = \frac{TP}{TP + FP + FN} \tag{7}$$

where TP stands for true positive (the area correctly classified as object), FP for false positive (area incorrectly classified as object), and FN for false negative (area incorrectly classified as background). Note that this accuracy definition is more stringent than many others discussed in Ref. 3.

Table 1. Performance of different algorithms for lung field (left and right) segmentation on two different data sets (along with the associated standard deviations). Note that CR is from computed radiography and JSRT is from film scan.

	QN	ShRAC	ASM
CR data set (137)	0.828 ± 0.065	0.892 ± 0.039	0.893 ± 0.060
JSRT data set (246)	0.870 ± 0.049	0.911 ± 0.027	0.930 ± 0.022

From Table 1, it is clear that ShRAC outperforms the QN optimization algorithm. For reference, we also computed the accuracy given by an implementation of the active shape model (ASM).[3,4] ASM obtains a slightly higher accuracy (1~2%) than ShRAC; however, this is at the cost of building two separate linear models for the left and right lung fields and for manually labeling a few important feature landmarks on each training image. Moreover, model selection (whether left or right) needs to be performed interactively (or automatically by a separate algorithm, if possible). We found that if the model selection fails, the segmentation by ASM would suffer significantly. This shows the flexibility of ShRAC where *nonlinear* variations of shapes can be competently handled by a single shape prior model. The slightly lower performance of ShRAC can also be attributed to its considerably weaker appearance model than ASM. However, we argue that a weak appearance model is not necessarily a drawback, especially in cases where the appearance of a particular object is not consistent in different images (e.g., images from imaging systems by different manufacturers).

3.2 Clavicle segmentation

To further demonstrate the effectiveness of the shape regularization, we also applied ShRAC to clavicle segmentation, which has not received much attention to date.[2] Segmenting the clavicle is extremely challenging because there are a large number of similar structures in the vicinity, including the first three ribs and shoulder joints. A learned, shape prior is critical for ShRAC to segment the correct object.

For shape learning, we manually segmented out 30 clavicles and aligned them similarly to lung fields. The only difference was that we no longer rotated the axis. The shape model was represented using 40 control points. The 30 aligned shapes were used to collect shape statistics for KSDE.

The segmentation is, again, a multi-scale iteration process consisting of DP searching, global regularization, and local regularization. During the search, we used a slightly different search grid from the one in lung field segmentation. Since the clavicle bone is a thin and long structure, searching along the outward normal direction of the contour is not efficient. Instead, we used a search direction perpendicular to the bone's mid-line for all the control points. Initially, the lung field centerlines and the spine centerline are found using the same algorithm that was used to initialize the lung field segmentation. We placed the mean shape (derived from all the training data) at the intersection of the spine and the line that passes through the two tips of the lung field centerlines. The segmentation stage runs in a multi-scale iterative fashion, starting from a large search grid step size that gradually decreases. The initial positions and the final segmentation results of two images are show in Fig. 8, along with the segmentation results when all the shape constraints were disabled. In the latter case, dynamic programming found a globally optimal contour that maximizes only the intensity difference, causing the contours to be attracted to other salient local structures.

Clavicles are long and thin structures unlike blobby lung fields; therefore, we are in the process of investigating a suitable quantitative metric for evaluating the segmentation performance.

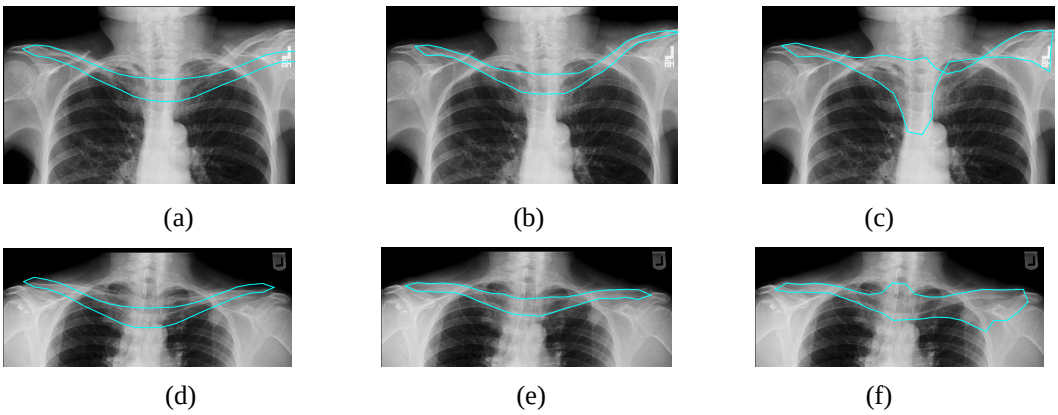


Figure 8: Comparison: (a and d) initial contours, (b and e) segmentation results from ShRAC, (c and f) segmentation results from DP without shape constraints.

4. DISCUSSIONS AND CONCLUSIONS

We propose shape regularized active contour (ShRAC), as a segmentation method that can incorporate nonlinear shape statistics and robustly segment anatomical structures in medical images. Our new contributions are:

1. Use kernel space density estimation as nonlinear shape priors in segmenting anatomical objects.
2. Propose two methods that can effectively integrate shape regularization from KSDE priors with dynamic programming in the active contour search.

The are many advantages to using ShRAC, including:

1. Use of shape priors for robust segmentation in the presence of noise and clutter
2. Use of dynamic programming to provide the ability to overcome local minima
3. No need for model selection to handle large shape variations
4. The weak appearance model in ShRAC makes it insensitive to appearance changes
5. Minimum human interaction is required in shape training. (Only shape statistics need to be learned. There is no need for manual identification of landmarks and object boundaries need not be highly accurate.)

In summary, the proposed ShRAC algorithm is able to learn nonlinear shape priors and enforce them during segmentation. ShRAC uses kernel space density estimation as the shape learning method. The learned shape priors are combined with dynamic programming to guide an active contour using global and local regularization algorithms. Experimental results on lung field segmentation and clavicle segmentation have shown that KSDE allows the object with large shape variations to be learned by a single shape prior model for segmentation. In addition, dynamic programming enables ShRAC to be robust and invulnerable to local minima. Finally, the two proposed shape regularization algorithms effectively enforce shape prior while taking advantage of dynamic programming to overcome local minima caused by noise and clutter. We believe that ShRAC represents a major step in the paradigm shift to object segmentation under nonlinear shape constraints.

ACKNOWLEDGMENTS

We thank our colleagues at Kodak, Wei Hao, Robert T. Gray, Mark Bolin, Shoupu Chen, Robert A. Senn, Hui Luo, Michael Heath, Teresa Levy, and Xiaohui Wang for help and discussions.

REFERENCES

1. L. Quekel, A. Kessels, R. Goei, and J. V. Engelshoven, “Miss rate of lung cancer on the chest radiograh in clinical practice,” *Chest*, vol. 115, no. 3, pp. 720–724, 1999.
2. B. V. Ginneken, B. M. H. Romeny, and M. A. Viergever, “Computer-Aided Diagnosis in Chest Radiography: A Survey,” *IEEE Trans. Medical Imaging*, vol. 20, no. 12, Dec. 2001.
3. B. V. Ginneken, A. F. Frangi, J. J. Staal, B. M. H. Romeny, and M. A. Viergever, “Active shape model segmentation with optimal features,” *IEEE Trans. Medical Imaging*, vol. 21, no. 8, Aug. 2002.
4. B. V. Ginneken, M. Bruijne, M. Loog, and M. A. Viergever, “Interactive Shape Models,” *SPIE Medical Imaging*, vol. 5032, pp. 1206–1216, 2003.
5. D. Cremers, F. Tischhäuser, J. Weickert, and C. Schnörr, “Diffusion Snakes: Introducing Statistical Shape Knowledge into the Mumford-Shah Functional,” *Int. J. Computer Vision*, vol. 50(3), pp. 295–313, Dec. 2002.
6. D. Cremers, T. Kohlberger, and C. Schnörr, “Nonlinear Shape Statistics via Kernel Spaces,” *German Conf. on Pattern Recognition*, Munich, vol. 2191, pp. 269–276, Sept. 2001.
7. D. Cremers, T. Kohlberge, C. Schnorr, “Shape statistics in kernel space for variational image segmentation,” *Pattern Recognition*, vol. 36, pp. 1929–1943, 2003.
8. D. Geiger, A. Gupta, L. A. Costa, and J. Vlontzos, “Dynamic programming for detecting, tracking, and matching deformable contours,” *IEEE Trans. on PAMI*, vol. 17, March 1995.
9. T. F. Cootes, C. J. Taylor, D. Cooper, and J. Graham, “Active shape models—their training and application,” *Computer Vision and Image Understanding*, vol. 61, no. 1, pp. 38–59, 1995.
10. MATLAB optimization toolbox: http://www.mathworks.com/access/helpdesk/help/toolbox/optim/2001.