External Force Modeling of Snakes using DWT for Texture Object Segmentation

Surya Prakash and Sukhendu Das

Visualization and Perception Lab,
Department of Computer Science and Engineering,
IIT Madras
Chennai-600 036, India.
E-mail: surya@cse.iitm.ernet.in, sdas@iitm.ac.in

Snakes, also known as active contours are extensively used in computer vision and image understanding applications. They are energy minimizing deformable contours that converges at the boundary of an object in an image. Deformation in contour is caused due to internal and external forces acting on it. Internal force is derived from the contour itself and external force is invoked from the image. Traditional active contours proposed by Kass et al. only work for the normal intensity images and fail to perform segmentation task in presence of texture. This limitation comes due to the limited ability of the external force present in the traditional snake, which uses directly image pixel's intensity information for its formulation. Here, we present a new external force modeling technique for snakes, which can work in presence of texture. It uses texture features for external force modeling, which are derived using discrete wavelet transform. To demonstrate our model, we use various synthetic and natural texture images.

Keywords: Snakes, active contours, texture, object segmentation, wavelet, DWT

1. Introduction

Snakes,[1] also known as active contours are extensively used in computer vision and image understanding applications. They are energy minimizing deformable contours that converge at the boundary of an object in an image. Deformation in contour is caused because of internal and external forces acting on it. Internal force is derived from the contour itself and external force is invoked from the image. The internal and external forces are defined so that the snake will conform to object boundary or other desired features with in the image. Snakes are widely used in many applications such as segmentation,[2] shape modeling,[3] edge detection,[1] motion tracking[4] etc. Active contours can be classified as either *parametric active contours*[1,5] or *geometric active contours*[6,7] according to their representation and implementation. In this work, we focus on parametric active contours, which synthesize parametric curves within image domain and allow them to move towards the desired image features under the influence of internal and external forces. The internal force serves to impose a piecewise continuity and smoothness constraint whereas external force pushes the snake towards salient image features like edges, lines and subjective contours. External force in the traditional snake is defined as the negative of the image gradient. In the presence of such external force, snake is attracted towards large image gradients i.e. towards the edges in the image. So if it is applied to the textured images, it will often get stuck on local texel (micro-units or cells of a texture) edges and converge at non-object boundary.

To overcome this effect, we here present a new class of external force for textured images, which we name as texture force. The snake in presence of texture force runs over the texture image surface and detects the object boundary of a texture surface, on a background texture. Texture force does not use directly the image pixel intensity values for its modeling. It considers the texture properties of the image.

2. Background

2.1. *Parametric Snake Model*

A traditional active contour is defined as a parametric curve $\mathbf{v}(s) = [x(s), y(s)]$, $s \in [0, 1]$, which minimizes following energy functional

$$E = \int_0^1 \frac{1}{2}(\alpha|\mathbf{v}^{'}(s)|^2 + \beta|\mathbf{v}^{''}(s)|^2) + E_{ext}(\mathbf{v}(s))ds \quad (1)$$

where, α and β are weighting constants to control the relative importance of the elastic and bending ability of snake respectively. $\mathbf{v}^{'}(s)$ and $\mathbf{v}^{''}(s)$ are the first and second order derivatives of $\mathbf{v}(s)$ and E_{ext} is derived from the image so that it takes on its smaller values at the feature of interest such as edges, object boundaries etc. For an image $I(x, y)$, where (x, y) are spatial co-ordinates, typical external energy is

defined as follows to lead snake towards step edges[1]

$$E_{ext} = -|\nabla I(x,y)|^2 \tag{2}$$

where, ∇ is gradient operator. A snake that minimizes E must satisfy following Euler equation

$$\alpha \mathbf{v}''(s) - \beta \mathbf{v}''''(s) - \nabla E_{ext} = \mathbf{0} \tag{3}$$

Eq. 3 can also be viewed as force balance equation

$$\mathbf{F}_{int} + \mathbf{F}_{ext} = \mathbf{0} \tag{4}$$

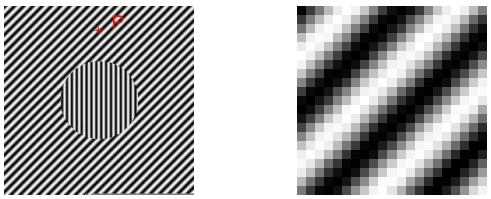
where, $\mathbf{F}_{int} = \alpha \mathbf{v}''(s) - \beta \mathbf{v}''''(s)$ and $\mathbf{F}_{ext} = -\nabla E_{ext}$. $\mathbf{F}_{int}$, the internal force, is responsible for stretching and bending and $\mathbf{F}_{ext}$, the external force, attracts snake towards the desired features in the image.

2.2. *Discrete Wavelet Transform and Scalogram*

The discrete wavelet transform (DWT) analyses a signal based on its content in different frequency ranges. Therefore, it is very useful in analyzing repetitive patterns such as texture. DWT decomposes a signal into different bands (approximation and detail) with different resolution in frequency and spatial extent. Let $I(x)$ be the image signal and $\psi_{u,s}(x)$ be a wavelet function at a particular scale, then signal filtered at point u is obtained by taking the inner product of the two $< I(x), \psi_{u,s}(x) >$. This inner product is called *wavelet coefficient* of $I(x)$ at position u and scale s.[8] *Scalogram*[9] of a signal $I(x)$ is the variance of this wavelet coefficient:

$$w(u,s) = \mathbf{E}\{| < I(x), \psi_{u,s}(x) > |^2\} \tag{5}$$

The $w(u,s)$ has been approximated by convolving the square modulus of the filtered outputs with a Gaussian envelop of a suitable width.[9] The $w(u,s)$ gives the energy accumulated in a band with frequency bandwidth and center frequency inversely proportional to scale.



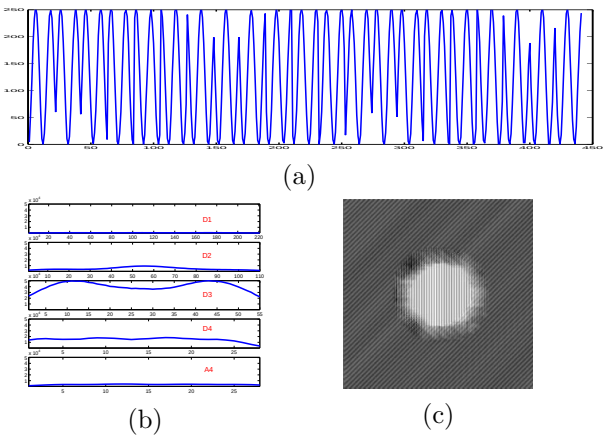
(a) (b)

Fig. 1. (a) Synthetic texture image, (b) Magnified view of the 21×21 window of texture cropped at point P (marked in RED color), shown in Fig. 1(a).

3. Texture Feature Extraction

In this section, we explain how the wavelet transform is used to extract texture features necessary for texture force estimation. It discusses the computational framework based on multi-channel processing. We use DWT-based dyadic decomposition of the signal to obtain texture properties. A simulated texture image shown in Fig. 1(a) is used to illustrate the computational framework with the results of intermediate processing.

Modeling of texture features at a point in an image involves two steps: scalogram estimation and texture feature estimation. To obtain texture features at a particular point (pixel) in an image, a $n \times n$ window is considered around the concerned point (see Fig. 1(b)). Intensities of the pixels in this window are arranged in the form of a vector of length n^2 whose elements are taken column wise from the $n \times n$ cropped intensity matrix. This intensity vector (signal), which basically represents the textural pattern around the pixel, is subsequently used in the estimation of scalogram.



(a) (b) (c)

Fig. 2. (a) 1-D texture profile of the texture window shown in Fig. 1(b), (b) Scalogram of the signal shown in Fig. 2(a), (c) Texture feature image for the image shown in Fig. 1(a).

Scalogram estimation: An input signal, obtained after arranging the pixels of $n \times n$ window as explained above, is used for the scalogram estimation. This signal is decomposed using wavelet filter. We use orthogonal Daubechies 2-channel (with dyadic decomposition) wavelet filter. Daubechies filter with level-L dyadic decomposition, yields wavelet

coefficients $\{A_L, D_L, D_{L-1}, .., D_1\}$ where, A_i represents approximation coefficient and D_i's are detail coefficients. The steps of processing to obtain scalogram from the wavelet coefficients are similar to that described in.[10,11] Fig. 2(b) presents an example of scalogram obtained for signal shown in Fig. 2(a) using level-4 DWT decomposition.

Texture feature estimation: Once the scalogram of the texture profile at a particular point is obtained, a post-processing step is carried out to eliminate non-significant bands of the scalogram and only significant bands are used for the further processing. This is done since only significant bands contain major texture features information. Removal of non-significant bands helps in removing the redundant information and making the computation fast. Let wavelet decomposition is done up to level-L and it gives following wavelet bands $B = \{A_L, D_L, D_{L-1}, .., D_1\}$. Let B_i is a i^{th} wavelet band in set B. We use following algorithm to determine significant and non-significant bands.

---

for each B_i **do**
  if variance for band $B_i \leq$ threshold
    B_i is non-significant band
  else
    B_i is significant band
  end-if
end-for

---

where threshold is decided empirically. Variance of band B_i is defined as $\text{var}(B_i) = \mathbf{E}[(B_i - \mu_i)^2]$ where μ_i is the mean of all wavelet coefficients belonging to band B_i. Once the estimation of scalogram and its significant bands is over, significant bands are used for texture feature estimation. Texture features are estimated from the "energy measure" of the wavelet coefficients of the significant bands. This texture feature is similar to the "texture energy measure" first proposed by Laws.[12]

Let for pixel k, D_k is the set of all significant bands and w is a wavelet coefficient in a band. Then the energy measure of pixel k is calculated to be the averaged l_1-norm as

$$E_k = \frac{1}{n}\left\{\sum_{X \in D_k}\sum_{w \in X} w\right\} \tag{6}$$

where, n is the sum of cardinalities of all the members of D_k. These energy measures for all pixels in an image constitute "texture feature image" which is further used in texture force modeling. Fig. 2(c) shows a texture feature image for the texture image shown in Fig. 1(a). Pixels belonging to the same texture region exhibit same energy level.

4. Modeling of Texture Force

This analysis is based on the gradient present in the texture feature image. Let for a given texture image $I(x, y)$, $F(x, y)$ be the texture feature image obtained as explained in previous section. The external energy of the snake based on the gradient present in the texture feature image can be defined as follows (similar to Eq. 2)

$$E_{ext}^{tex} = -|\nabla F(x, y)|^2 \tag{7}$$

As done in Eq. 4, texture force (external force), which causes the change in this energy (i.e. E_{ext}^{tex}), can be defined as follows

$$F_{ext}^{tex} = -\nabla E_{ext}^{tex} \tag{8}$$

To find the object boundary, active contour deforms so it can be represented as the time varying curve $\mathbf{v}(s, t) = [\mathbf{x}(s, t), \mathbf{y}(s, t)]$ where $s \in [0, 1]$ is arc-length and $t \in R^+$ is time. Dynamics of the contour in presence of texture force can be governed by following equation

$$\gamma \mathbf{v}_t = \mathbf{F}_{int} + \mathbf{F}_{ext}^{tex} \tag{9}$$

where, $\mathbf{v}_t$ is the partial derivative of $\mathbf{v}$ *w.r.t.* to t, $-\gamma\mathbf{v}_t$ is the damping force and γ being an arbitrary non-negative constant. $\mathbf{F}_{int}$ and $\mathbf{F}_{ext}^{tex}$ are internal and texture forces respectively. The contour comes to rest when the net effect of the damping, internal, and texture force reaches to zero, which eventually happens when deforming contour reaches the texture object boundary. The texture force developed here pushes active contours towards texture object boundary.

5. Experimental Results

To get the boundary of a particular object using active contours in presence of texture, a contour is initialized near the desired object. Contour is then allowed to deform towards the object boundary until it

latches around the object. In case of textured images, object boundary is identified as the point where texture property changes i.e. where two texture regions meet. In texture surface, snake in presence of texture force stops moving as it gets different texture region. For a snake to stop at the texture boundary, net effect of the damping, internal and external forces should be zero for all snake points at the object boundary. To demonstrate the performance of the snake in the presence of texture force, various kinds of synthetic and natural textures are used. We have used Daubechies 8-tap 2-channel filter for DWT decomposition.

The first example is of a texture image (Fig. 3) composed of two textures taken from a widely used Brodatz photographic album.[13] Contour is initialized around the central texture and is allowed to shrink in presence of texture force. Snake took 14 iterations to converge at the central texture boundary. Texture features at snake points are estimated by taking a 15×15 window at each point. DWT decomposition was done up to level-4. Texture feature image for this test image in shown in Fig. 4(a). The resulting segmentation is shown in Fig. 4(b), where the identified texture object boundary is shown in dark black color.

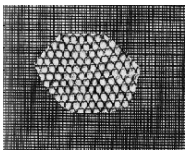
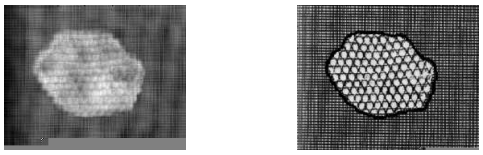


Fig. 3. Texture image composed of two Brodatz textures.[13]



(a) (b)

Fig. 4. (a) Texture feature image for the texture image shown in Fig. 3, (b) Final Segmentation result of Fig. 3. Dark black contour shows the estimated boundary of the central texture.

We present another segmentation result for an image (Fig. 5) composed of two Brodatz textures. Contour convergence to the central texture boundary, in presence of texture force, took 11 iterations. Texture features at contour points are estimated by taking a 13×13 window. DWT decomposition was done up to level-4. Texture feature image for this test image is shown in Fig. 6(a). The resulting segmentation is shown in Fig. 6(b), where the identified texture object boundary is shown in dark black color.

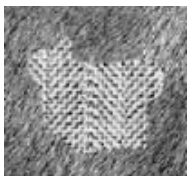
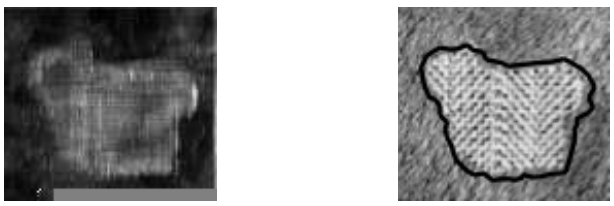


Fig. 5. Texture image composed of two Brodatz textures.[13]



(a) (b)

Fig. 6. (a) Texture feature image for the texture image shown in Fig. 5, (b) Final Segmentation result of Fig. 5. Dark black contour shows the estimated boundary of the central texture.

Fig. 7. Natural real life test image of zebra.

(a) (b)

Fig. 8. (a) Texture feature image of zebra (shown in Fig. 7), (b) Final Segmentation result for zebra. Dark black contour shows the estimated boundary of zebra

Fig. 7 shows a natural real life test image of zebra. Contour convergence to the boundary of zebra,

in presence of texture force, took 15 iterations. Texture features at contour points are estimated by taking a 11×11 window. DWT decomposition was done up to level-4. Texture feature image of zebra is shown in Fig. 8(a). The resulting segmentation is shown in Fig. 8(b), where the identified boundary of the zebra is shown in dark black color.

(a) (b)

Fig. 9. Comparison of segmentation results of zebra image, (a) Segmentation result obtained using proposed technique, (b) Segmentation result obtained in.[14] Dark black contour shows the estimated boundary of zebra in both the cases.

6. Conclusion

In this paper, we have introduced a new external force for snakes, which we call as texture force. Snake, in presence of texture force, can be used for the texture object segmentation. To model texture force, first texture features are estimated using wavelet decomposition which are further used in texture force modeling. Texture force is subsequently used in parametric snakes for texture object segmentation. Main novelty of this study is in the representation of the texture features and the modeling of texture force based on them. We validate our model with a few synthetic and natural texture images. Results obtained using proposed technique are quite satisfactory. In Fig. 9, we compare our segmentation result of zebra with the segmentation result obtained by Sagiv et al. in[14] for the same. The result obtained by proposed technique (Fig. 9(a)) is comparable with the result obtained in[14] (Fig. 9(b)). Since proposed segmentation technique uses parametric snake, it is computationally less expensive compare to the technique presented in[14] which uses geodesic active contour.

References

1. M. Kass, A. Witkin and D. Terzopoulos, *International Journal of Computer Vision* **1**, 321(January 1988).
2. F. Leymarie and M. D. Levine, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **15**, 617 (1993).
3. D. Terzopoulos and K. Fleischer, *The Visual Computer* **4**, 306 (1988).
4. D. Terzopoulos and R. Szeliski, Tracking with Kalman snakes, in *Active Vision*, eds. A. Blake and A. L. Yuille (MIT Press, Cambridge, MA, 1992) pp. 3–20.
5. L. D. Cohen, *CVGIP: Image Understanding* **53**, 211 (1991).
6. V. Caselles, F. Catte and T. Coll, *Numerische Mathematik* **66**, 1 (1993).
7. V. Caselles, R. Kimmel and G. Saprio, *International Journal of Computer Vision* **22**, 61 (1997).
8. S. Mallat, *A Wavelet Tour of Signal Processing* (Academic Press, 1999).
9. M. Clerc and S. Mallat, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **24**, 536(April 2002).
10. T. Greiner and S. Das, *Automatisierungstechnik* (2005).
11. S. G. Rao, S. Das, T. Greiner and M. Kalra, *Accepted in IEE VIE Conference, Bangalore, September 26-28* (2006).
12. K. Laws, Textured image segmentation, PhD thesis, (Dept. of Electrical Engineering, University of Southern Californiai, 1980).
13. P. Brodatz, Textures: A photographic album for artists and designers (1966).
14. C. Sagiv, N. Sochen and Y. Zeevi, *IEEE Transactions on Image Processing* **1**, 1(February 2004).