

# Wavelet-based Active Contour Model for Object Segmentation and Tracking in Video Sequences

Jen-Chang Liu<sup>††</sup>, Wen-Liang Hwang<sup>†</sup>, Ming-Syan Chen<sup>‡</sup>, Jin-Wu Tsai<sup>\*</sup>, and Chi-Hung Lin<sup>\*</sup>

Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan<sup>‡</sup>

Institute of Information Science, Academia Sinica, Nankang, Taipei, Taiwan<sup>†</sup>

Institute of Microbiology and Immunology, National Yang-Ming University, Taipei, Taiwan<sup>\*</sup>

## Abstract

We propose an integrated wavelet-based framework of the active contour model (*snake*) for segmentation and motion tracking of deformable objects in video sequences. The input image frame is represented by means of wavelet transform. First, the moduli of wavelet transform coefficients are used in a multi-resolution motion estimation process to find the initial snake contour in the current frame. The presented multi-resolution motion estimation methods allow a larger movement of the tracked object than does traditional image-based motion estimation. Secondly, the wavelet transform modulus at each scale is considered in the energy function of the snake model. The snake computation is based on a coarse-to-fine scale continuation method. Application of the proposed methods to biological cell tracking is demonstrated in experiments.

Address : Institute of Information Science, Academia Sinica, Taiwan

e-mail: whwang@iis.sinica.edu.tw

# 1 Introduction

The problems of object segmentation and object tracking in video sequences have become more important in recent years because of rapid development of advanced video compression standards, such as MPEG-4[1] and MPEG-7[2]. These modern video compression standards support image object description; therefore, object segmentation and object tracking are required in the preprocessing steps. Many algorithms have been proposed to solve these problems. Among them, we are especially interested in the method which employs the active contour model(*snake*) [6]. Not only can it solve object segmentation and tracking problems simultaneously [6][5][7], but it also can employ high-level visual knowledge as an aid by means of an interactive process. There are two important issues in applying the snake model to video sequences: the generation of an initial contour in each frame and the design of external forces. These two issues correspond to the problems of object tracking and object segmentation, respectively. In this paper, we propose an integrated wavelet-based snake framework for object tracking and segmentation in video sequences.

Snake is basically an energy minimization model, which seeks to minimize the energy of a contour spline in an image. The energy of the contour includes the energy caused by the external forces, internal forces, and other constraint forces. The design of external forces determines the goal of energy minimization. In the application of object segmentation, the design of external forces must produce low energy values at the object boundaries or edges. Once a reasonably good initial contour is chosen in an image frame, the energy minimization process will attract the contour spline to the object boundary. In early work [8], the gradient of an image was used as the external force to attract a snake to the object boundary. Later, a hierarchical filtering method, also known as the *scale-space continuation method* [7][8][9], was often used to find the best solution of the snake. This continuation method is a process which tracks the best solution from a coarse scale to finer scales in a scale-space representation of an

image; thus, it has the advantage of robustness with respect to image noises in the snake computation process.

In this paper, we extend the idea of the gradient-based scale-space continuation method to wavelet-based processing of the active contour model. The modulus of wavelet transform coefficients is taken as the external force. A few researches [3][4] also applied wavelet transform to the active contour model, but they did so in different ways. The authors in [3] proposed a wavelet-frame based image force model in the energy function, but they used the dynamic programming approach instead of the continuation method in snake computation. The authors in [4] computed a multi-scale edge representation using wavelet transform and fitted their defined wavelet snake contour to the edge representation. They applied a gradient descent algorithm to update the snake contour.

To deal with the problem of object tracking, a simple method is to use the snake contour of the previous frame as an initial snake for object segmentation in the current frame [7]. However, when the object moves quickly, this method tends to result in a bad initial contour, leading to a poorer snake in the current frame. Therefore, we apply a modified approach to multi-resolution motion estimation (MRME)[10][12] to estimate the motion vector at each snake pixel, thus generating an estimated initial contour for the current frame. Our method differs from MRME in that we use the modulus of wavelet transform coefficients rather than all the subband coefficients of the wavelet transform. Therefore, our method concentrates on edge information rather than wavelet coefficients at each subband for motion estimation. Finally, an empirical method of cycle-spinning-like motion estimation is proposed to improve the proposed algorithm.

Our integrated wavelet-based active contour framework for object segmentation and tracking is shown in Fig. 1. This proposed framework can be embedded in the wavelet-based video coding system [11] for the purpose of object segmentation and tracking.

Thus, one-pass of wavelet transform can provide all necessary information for coding, object segmentation, and object tracking in this integrated framework. The details will be given in the following sections. In Section 2, we describe the application of wavelet transform in the active contour model. In Section 3, we discuss issues related to motion estimation in object tracking. In Section 4, application of biological cell tracking is demonstrated in experiments. Conclusions are given in Section 5.

## 2 The wavelet-based active contour model

In this section, we will first review the snake model and the scale-space continuation method. Then we will explain our proposed wavelet-based active contour model.

### 2.1 Snake and the scale-space continuation method

A snake describes a deformable curve  $v(s, t)$ , where parameter  $s$  is the spatial index and  $t$  is the time index. In an image plane, this deformable curve is a function of the coordinate variables  $x$  and  $y$ :

$$v(s, t) = (x(s, t), y(s, t)) : s \in \Omega, t \in T, \quad (1)$$

where  $\Omega$  and  $T$  are defined open intervals. The energy function of a snake is defined as [6][7]

$$E(v) = \frac{1}{2} \int_{\Omega} [E_{int}(v) + E_{ext}(v) + E_{con}(v)] ds, \quad (2)$$

where  $E_{int}$  is the internal force of the snake,  $E_{ext}$  is the external force of the snake caused by the image, and  $E_{con}$  is the additional constraint forces. Details of the formulas and designs of these forces can be found in [6][7]. The snake model undergoes an energy minimization process when a curve  $v^*(s)$  with minimal energy  $E(v^*)$  is calculated. The key requirement for the curve  $v^*(s)$  to be the object contour will depend on proper design of the image force  $E_{ext}$ . For example, the most common method used to find an

edge contour in an image is to set  $E_{ext} = -|\nabla I(x, y)|$ , where  $I(x, y)$  is the gray-level intensity of the image. The snake will be attracted to places with large image gradients, which correspond to possible image edges.

Because most images contain a lot of noise, directly applying the snake model to the original images often produces unstable results. The idea of the scale-space continuation method is to calculate the snake in a coarsely smoothed image; then, the result of the snake at the coarse scale is used as an initial contour in a finely smoothed image to compute a fine-tuned snake. This fine-tuning process is repeated. Implementation of the gradient-based scale-space continuation method presented in [7] is illustrated in Fig. 2.

As shown in Fig. 2, the original image is first filtered through a family of Gaussian filters with different resolutions. Then, a differentiation filter, such as the Sobel filter, is applied to these Gaussian filtered images to produce approximations of the gradients of the Gaussian smoothed images. The continuation method starts from the gradient of the Gaussian smoothed image at the coarsest scale. After the snake at the coarsest scale is found, it is taken as an initial contour for the following finer scale. The process is repeated until the finest scale is reached. The scale-space continuation method provides robustness with respect to image noise in real applications.

## 2.2 Proposed design of the wavelet-based snake

In this section, we will generalize the idea of the gradient-based scale-space continuation method to fit the wavelet-based framework. The multi-scale structure of the continuation method can be achieved by means of wavelet transform. In fact, it has been shown [13] that the first derivatives of a well-defined family of Gaussian filters are equivalent to the corresponding wavelet transform coefficients multiplied by a scaling constant.

Let the family of Gaussian filters be properly chosen so as to satisfy the 2-D dilation

equation:

$$\theta_s(x, y) = \frac{1}{s^2} \theta\left(\frac{x}{s}, \frac{y}{s}\right). \quad (3)$$

We define the 2-D wavelet functions in the  $x$ - and  $y$ - directions as

$$\psi^1(x, y) = \frac{\partial \theta(x, y)}{\partial x}, \psi^2(x, y) = \frac{\partial \theta(x, y)}{\partial y}; \quad (4)$$

then the wavelet transforms of image  $I(x, y)$  in the  $x$ - and  $y$ - directions at scale  $s$  are

$$W_s^1 I(x, y) = I * \psi_s^1(x, y), W_s^2 I(x, y) = I * \psi_s^2(x, y). \quad (5)$$

It can be shown [13] that

$$\begin{pmatrix} W_s^1 I(x, y) \\ W_s^2 I(x, y) \end{pmatrix} = s \vec{\nabla} (I * \theta_s)(x, y). \quad (6)$$

Therefore, the above equation implies that applying wavelet transform is equivalent to applying both smoothing and gradient operations.

In [14], it was shown that fast implementation of (6) can be achieved when  $s$  is dyadic by filtering alternatively through a low-pass filter(L) and a high-pass filter(H) accordingly (see Fig. 3(a)). The fast implementation of (6) has a structure similar to the analysis part of a filter bank except that the down-sampling operation is not applied.

Since  $-|\nabla I(x, y)|$  is often used as the external energy, we can define the external energy  $E_{ext}$  at scale  $s$  as the negative of the modulus of wavelet transform at scale  $s$

$$E_{ext}^s(x, y) = -\sqrt{|W_s^1 I(x, y)|^2 + |W_s^2 I(x, y)|^2}. \quad (7)$$

The above definition of external energy together with the continuation method in the wavelet domain serves to generalize the gradient-based scale-space continuation method. The architecture of our proposed wavelet-based snake model is shown in Fig. 3(a). Compared to the original gradient-based scale-space continuation method, our

proposed model applies the theory of wavelet transform in analysis and is more flexible in terms of the choice of filter.

Fig. 3(b) shows an alternative design to that shown in Fig. 3(a) based on decimation of the wavelet coefficients. Because the size of the snake pixels (*snaexels*) determines the computational complexity of the snake model, it is possible to reduce the computational complexity in the down-sampled wavelet domain. A simple implementation is proposed as follows. First, the initial snaexels of the input frame are down-scaled and down-sampled to obtain an initial snake in the coarsest scale. After the final snake in this scale is obtained, we up-scale and up-sample (by interpolation) the snaexels of the snake and propagate them to the next fine scale. We repeat this process until the finest scale is reached. Although the computational complexity is reduced in this down-sampling framework, we lose computational accuracy. The solution of the snake is not reliable at coarse scales due to the low resolution of the snaexels and the modulus images. Thus, we will present experiments in Section 4 conducted using the framework shown in Fig. 3(a). However, the design shown in Fig. 3(b) still provides an interesting choice for applications where computational complexity is more critical than computational accuracy.

### 3 Motion estimation in contour tracking

Given a good initial contour, the snake will converge to the object boundary in an image during the energy minimization process. In a sequence of images, a target object can have both global movement and local deformations. If the object moves or deforms slowly between consecutive frames, the resulting snake at frame  $(i - 1)$  can be used as a reasonably good initial contour at frame  $i$ . Then, the object contour at frame  $i$  can be found using the snake model with this initial contour. However, the constraints of *small motion and deformations* are too restrictive to be applied to all kinds of image sequences. Thus, motion estimation for the initial contour at each frame is necessary

when an object moves rapidly or deforms drastically between frames.

We propose to match the snake contour pixels (*snaxels*)  $v[s, i - 1]$  at frame  $(i - 1)$  to the estimated contour points  $v[s, i]$  at frame  $i$  by means of motion estimation. Many motion estimation methods have been proposed in the literature [16]. Among them, block-based methods are commonly used, including the full search, three-step search, hierarchical motion estimation [17] and multi-resolution motion estimation(MRME) methods[10]. The full search method is *optimal* but the least efficient one, while the others are sub-optimal but faster methods. These methods usually involve a trade-off between computational complexity and estimation accuracy. Since we use the wavelet modulus in the active contour model, the computational cost can be reduced if we employ a motion estimation method in the same framework. Thus, we propose an MRME method based on using the wavelet modulus for object tracking. We will show that by adopting a concept similar to cycle-spinning[15], we can improve the precision of the proposed method. Before introducing our methods, we will review the MRME method first.

### 3.1 The multi-resolution motion estimation method

We will define the problem of block-based motion estimation formally. Let  $I_i(x, y)$  be the pixel at coordinate  $(x, y)$  in frame  $i$ , and let  $V = (v_x, v_y)$  be the motion vector. The block-based motion estimation problem is to match a block centered at  $(x, y)$  in frame  $i - 1$  with the block centered at  $(x + v_x, y + v_y)$  in frame  $i$  with the minimum difference. Using the minimum mean absolute difference(MAD) criterion, the block-based motion estimation problem can be formulated as follows:

$$\min_{v_x, v_y} \frac{1}{(K + 1)^2} \sum_{i=-K/2}^{K/2} \sum_{j=-K/2}^{K/2} |I_{i-1}(x + i, y + j) - I_i(x + v_x + i, y + v_y + j)| \quad (8)$$

*subject to*  $-R \leq v_x \leq R, -R \leq v_y \leq R,$

where  $(K + 1) \times (K + 1)$  is the block size, and  $(2R + 1) \times (2R + 1)$  is the size of the search region. Both the block size and the size of the search region determine the computational complexity of motion estimation.

The full search method exhaustively searches all the pixels in the search region to find a matching block with the minimum MAD. Thus, the full search method is optimal but the least efficient one. The idea of multi-resolution motion estimation(MRME) [10] is to improve the performance in terms of computational complexity. Fig. 4(a) shows the typical subband structure of wavelet transform. LL, LH, HL, and HH represent the combination of low-pass and high-pass filtering in the  $x$ - and  $y$ - directions respectively, and their subscripts denote the indices in the scale space. Fig. 4(b) shows the estimated motion vectors in the subbands. Note that the motion vectors in the same subband orientation at different scales are not related by simple scaling due to the effects of down-sampling and the non-shift-invariant nature of wavelet transform. In one implementation, MRME starts matching by the coarsest subbands within the base search region  $p \times p$ , and the estimated motion vectors are then propagated through finer subbands and refined within a proper(say,  $r \times r$ ) search region as follows:

$$V_{i,j}(x, y) = 2V_{i,j+1}(x, y) + \Delta_{i,j}(x, y), \quad (9)$$

where  $\Delta_{i,j}$  is the refinement term,  $i$  denotes the subband orientation,  $i = 1, 2, 3$ , and  $j$  denotes the scale index,  $j = L, \dots, 1$  (from coarse to fine, where  $L$  represents the total decomposition levels). Comparing the full search method with MRME, the matched block size is constant  $(K + 1) \times (K + 1)$  in the full search method, while it is  $2^{-j}(K + 1) \times 2^{-j}(K + 1)$  at scale  $j$  in MRME; the size of the search region is  $(2R + 1)^2$  in the full search method, while it is  $p^2 + 3r^2L$  in MRME if refinement is done in all subbands and orientations. Computational complexity is reduced in MRME due to the reduction in the block size and search region size. Note that the base search region determines the maximally detectable motion in the image, that is,  $p2^L \times p2^L$ . To increase the maximally detectable motion in the image, a smaller increase in the size of the search

region is required for MRME than for the full search method. Thus, it is advantageous to apply MRME to image sequences when there are large object movements.

### 3.2 Multi-resolution motion estimation using the wavelet modulus

In this section, we will propose a motion estimation method which uses the wavelet modulus. Our method is similar in spirit to the MRME algorithm and to the hierarchical block matching algorithm (HBM) [15]. This means that we use both the down-sampled wavelet structure and block matching in our algorithm. However, our algorithm is not the same as the MRME algorithm since we are matching a snake contour on the wavelet modulus image at each scale instead of all the wavelet coefficients at each subband. MRME must produce respective motion vectors in all subbands for the purpose of coding, while we need one motion vector for each snaxel. Our algorithm is also different from the HBM algorithm. In HBM algorithm, only the low-passed (or smoothed) images are used for matching, while in our algorithm, we take advantage of the wavelet modulus and concentrate on the motion of edges rather than other image contents.

We start from estimating the snaxels in the coarsest smoothed image (say at scale  $2^L$ ) of frame  $k$  based on the snaxels at the corresponding scale in frame  $k - 1$ . Let  $S_{0,i}^{k-1} = (x_i^{k-1}, y_i^{k-1})$  be the  $i$ -th snaxel at frame  $k - 1$ . Then, the corresponding snaxel at scale  $2^L$  is  $S_{L,i}^{k-1} = \frac{S_{0,i}^{k-1}}{2^L} = (\lfloor \frac{x_i^{k-1}}{2^L} \rfloor, \lfloor \frac{y_i^{k-1}}{2^L} \rfloor)$ , where  $\lfloor x \rfloor$  is the largest integer no less than  $x$ .

Centered at each snaxel  $S_{L,i}^{k-1}$ , we find the matching snaxel at scale  $2^L$  in frame  $k$  by means of block-based motion estimation with an equation similar to that in (8) except that we concentrate only on each snaxel  $(x_s, y_s)$ . Let the most smoothed images of frames  $k - 1$  and  $k$  be, respectively,  $S_{k-1}^L$  and  $S_k^L$ , where  $L$  is the scale index. Then,

we find  $v_x^*$  and  $v_y^*$  to be

$$\min_{v_x, v_y} \frac{1}{(K+1)^2} \sum_{m=-\frac{K}{2}}^{\frac{K}{2}} \sum_{n=-\frac{K}{2}}^{\frac{K}{2}} |S_{k-1}^L(x_s + m, y_s + n) - S_k^L(x_s + v_x + m, y_s + v_y + n)|,$$

where  $v_x, v_y$  are over a pre-defined search region at the coarsest scale. This region is chosen so as to be large enough for object motion and deformations.

After the initial motion vectors are obtained, we then start to use wavelet modulus images at finer scales to refine the motion vectors. Motion vectors are migrated to the next finer scale by scaling each of them by 2, followed by a refinement process. The refinement process matches based on the wavelet modulus images in the neighborhoods of the snaxels at the same scale of these two consecutive frames. The scaling and refinement processes are repeated until the finest scale is reached. Finally, the resultant motion vectors are refined again between the original image frames. Our algorithm is summarized as follows.

**Algorithm.** Multi-resolution motion estimation using the wavelet modulus

**Step 1** Let the snaxels at frame  $k-1$  to be  $S_{0,i}^{k-1}$ , where  $i = 1, \dots, J$  is the snaxel index.

**Step 2** The snaxels at the coarsest scale  $2^L$  of frame  $k-1$  are  $S_{L^c,i}^{k-1} = \frac{S_{0,i}^{k-1}}{2^L}$ .

**Step 3** Estimate the motion vectors  $\{V_{L^c,i} | i = 1, \dots, J\}$  at snaxel positions  $S_{L^c,i}^{k-1}$  between the smoothed images at scale  $2^L$  of frames  $k-1$  and  $k$ .

**Step 4** Refine the motion vectors  $\{V_{L^c,i} | i = 1, \dots, J\}$  at snaxel positions  $S_{L^c,i}^{k-1}$  between the wavelet modulus images at scale  $2^L$  of frames  $k-1$  and  $k$ :

$$V_{L,i} = V_{L^c,i} + \Delta_{L,i}.$$

**Step 5** Go to the next finer scale  $2^j$ . Estimate the refinement term  $\Delta_{j,i}$  of the motion vector  $V_{j+1,i}$  at snaxel positions  $S_{j,i}^{k-1} = \frac{S_{0,i}^{k-1}}{2^j}$  by matching wavelet modulus images at the scale  $2^j$  of frame  $k-1$  and  $k$ . Then, the refined motion vector is

$$V_{j,i} = 2V_{j+1,i} + \Delta_{j,i}.$$

**Step 6** Repeat Step 5 until the finest scale is reached.

**Step 7** Refine the motion vectors  $\{V_{1,i}|i = 1, \dots, J\}$  at snaxel positions  $S_{0,i}^{k-1}$  using the image frames  $k - 1$  and  $k$ :

$$V_{0,i} = 2V_{1,i} + \Delta_{0,i}.$$

**Step 8** The estimated initial snaxel positions at frame  $k$  are  $\{S_{0,i}^k|i = 1, \dots, J\}$ , where

$$S_{0,i}^k = V_{0,i} + S_{0,i}^{k-1}.$$

Our algorithm can be explained by using power series expansion of the motion vectors (only up to pixel precision). For a motion vector  $V = (v_x, v_y)$  of a snaxel, we can expand  $v_x$  as follows:

$$v_x = a_N 2^N + a_{N-1} 2^{N-1} + \dots + a_1 2^1 + a_0, \quad a_i \in \{0, 1\},$$

and we can expand  $v_y$  similarly. Since we have down-sampling by 2 at each wavelet decomposition, the motion vector  $v_x$  at scale  $2^L$  becomes  $v_x^L = a_N 2^{N-L} + a_{N-1} 2^{N-1-L} + \dots + a_L$ . By choosing a sufficiently large searching region, we are able to estimate  $v_x^L$ , that is, to find the coefficients  $a_i$ 's for  $i = N, N - 1, \dots, L$  in step 3. By going to the next finer scale and multiplying the current motion vector by 2, we have  $2 \times v_x^L$ , which is equal to  $a_N 2^{N-L+1} + a_{N-1} 2^{N-L} + \dots + a_L 2^1$ . The refinement process started in Step 4 searches for  $a_{L-1}$  to obtain  $v_x^{L-1}$  in a proper neighborhood region. By repeating Step 5 and modifying the motion vector at step 7, we obtain the estimated  $v_x$ . A similar approach can be applied to  $v_y$ .

The computational complexity of the proposed multi-resolution motion estimation method using the wavelet modulus is equal to that of the MRME method. An illustration of the proposed method is shown in Fig. 5, where 3 scales of wavelet decomposition are applied. The flow chart of the proposed method is shown in Fig. 6.

### 3.3 The cycle-spinning-like motion estimation method

The main problem of applying motion estimation in the wavelet domain is the lack of translation invariance of the wavelet basis. A shift in the image domain cannot lead to corresponding number of shifts in the wavelet domain at different scales. One approach to suppressing such artifacts is termed “cycle-spinning” [15], which averages out the translation dependence. We propose a cycle-spinning-like motion estimation method to improve the performance of motion estimation in the wavelet domain. Let  $S_h$  denote the circulant shift by vector  $h$ , and let  $WMV$  denote the motion estimation in the wavelet domain. Cycle-spinning-like motion estimation can be formulated as follows:

$$\underset{h \in H}{Average} \quad S_{-h}(WMV(S_h(I))), \quad (10)$$

where  $H$  is the range of shifts and  $I$  denotes the original image. Although the proposed motion estimation algorithm is more computationally expensive than that described in the previous section, it provides improved performance in motion estimation in the wavelet domain. This is the trade-off between estimation performance and computational complexity. The improvement achieved with cycle-spinning-like motion estimation over the multi-resolution motion estimation using the wavelet modulus will be demonstrated through experiments in Section 4.3.1. Further studies on the performance of the cycle-spinning-like method in motion estimation are currently underway.

## 4 Application to biological cell tracking

In the biological sciences, it is important to obtain information about the movements of living cells, such as shape deformation, dynamics, and the path of motion [19]. Quantitative analysis of related data is required, but it is impractical for humans to do this. Thus, an automatic approach to tracking and analyzing the cell motions in video sequences is required. Several methods have been proposed in the literature [7][18].

We will present the results for biological cell tracking obtained using our proposed algorithms in the following subsections.

#### 4.1 Experimental data

Two representative image sequences of biological cells obtained as in [20] were used in the following experiments. In the first image sequence as shown in Fig. 8(a), several Chinese Hamster Ovary, CHO, cells were visualized under a bright field microscope, and detected and recorded by a CCD camera. Two aspects need to be considered when analyzing this kind of image sequence. First, the gray-level values of most of the different parts of these cells are very similar to that of the background, making it very difficult to identify the cell margin using conventional methodologies, such as thresholding. Secondly, in our study the observed Brownian motion was very fast, and could be resolved only using programs that have high temporal resolution.

The second movie, shown in Fig. 8(b), was taken when a CHO cell (B) was trapped and then moved (arrow) using the laser tweezers so that it made contact with another CHO cell (A). This sequence was observed using typical differential interference contrast (DIC) microscopy. The "half dark" (arrowhead) and "half light" (double arrowhead) shadow of the cell margin posed another difficulty for tracking these cells using conventional algorithms. The motion analyzed here represented a typical guided non-random motion. Note that the bar in the lower-left corner of Fig. 8(b) equaled 20 micrometers.

#### 4.2 Cell segmentation

First, we applied our proposed wavelet-based active contour model to the biological cell image. The wavelet basis that we used in the following experiments was a quadratic spline of compact support as defined in [13]. We used 4-scale decomposition, and used wavelet modulus images at scale  $2^4$ ,  $2^3$  and  $2^2$  for snake computation. The scale  $2^1$

(finest scale) wavelet modulus was discarded because it contained much noise.

Fig. 9(a) shows an image of the leftmost cell in Fig. 8(a), and Fig. 9(b)-(d) shows its wavelet transform modulus at different scales. The image was smoother at the coarser scale; thus, the noise was smoothed and the object boundary can be seen more clearly in the modulus image. Our algorithm segmented a rough cell boundary at the coarsest scale using the wavelet-based snake and then refined the contour at finer scales continuously. Fig. 10 shows a 3-D plot of Fig. 9(c), where the height is the negative value of the gray-level intensity, which corresponds to the external force. The energy minimization process forced the snake contour to move to the valley, which corresponded to the boundary of the cell.

Calculation of energy minimization was based on the discretized version of (2), and the finite difference method was applied in our implementation. The solution was derived using an iterative process in which linear equations were solved through inversion of a pentadiagonal banded matrix [6][7]. The terminating condition for snake convergence was based on the *steady-support* criterion [7], which sought to minimize the averaged energy

$$E_{length}(v) = \frac{\int_{\Omega} E_{ext}(v) ds}{\int_{\Omega} |v_s| ds} \quad (11)$$

along the snake. Fig. 11(a) shows the snake contour found at scale 4, where 'o' denotes the initial snaxels and '+' denotes the converged snaxels. Fig. 11(b) shows the plot of  $E_{length}(v)$  as a function of the number of iterations. Note that the necessary number of iterations decreased at finer scales.

### 4.3 Cell tracking

#### 4.3.1 Comparison of the proposed motion estimation methods

We conducted experiments with our proposed multi-resolution motion estimation method using the wavelet modulus and with the cycle-spinning-like motion estimation method.

Fig. 12(a) shows a  $256 \times 256$  image cut from the leftmost cell shown in Fig. 8(a). We manually created a 10-pixel shift both upward and to the left of Fig. 12(a). The original image was divided into  $8 \times 8$  blocks, and we applied our motion estimation methods to calculate the motion vectors. An estimated image was reconstructed from the shifted image and the motion vectors. The mean square error(MSE) between the estimated image and the original image was calculated to compare the performance of the methods. Figs. 12(b) and (c) showed the differences between the estimated images obtained using the two methods and the original image. Fig. 12(c) seems to show more noise than Fig. 12(b), and this phenomenon reveals the effects of averaging. The MSE of the multi-resolution motion estimation obtained using the wavelet modulus was 13.78, while the MSE of the cycle-spinning-like motion estimation was 5.35. It is clear that the cycle-spinning-like motion estimation method performed better than the multi-resolution motion estimation method using the wavelet modulus, at the cost of increased computational time.

### 4.3.2 Results of cell tracking

From the image sequence shown in Fig. 8(a), we took a  $256 \times 256$  window centered around the leftmost cell. This cell exhibited small movement in this windowed image sequence. We intentionally added a random shift ranging from  $-8$  to  $+8$  to each frame in this image sequence and applied our object segmentation and tracking algorithm. Figs. 13(a) and (b) show the intermediate tracking results, where the neighboring image frames are overlapped and the extracted contours are plotted. In Fig. 13(c), the added random movement and the average of the estimated motion vectors of snaxles are shown together. The average estimated motion vector was obtained by averaging the motion vectors estimated at each snake pixel. We observed that the estimated motion vector matched the added random motion reasonably well, and that the difference between them could be compensated for in the snake computation process. In this example, the average error of motion vector estimation was  $(-0.2, 0.1526)$ . Note that we employed

the multi-resolution motion estimation method using the wavelet modulus in all the experiments described in this section.

Fig. 14 shows the results of tracking a cell with a large amount of movement in the image sequence with 20 frames. The motion analyzed here represents a typical guided non-random motion. As shown in Fig. 14(d), we were able to track the motion of the lower cell very well.

## 5 Conclusions

We have proposed an integrated framework for a wavelet-based active contour model. This framework has two main parts: object segmentation and object tracking. In object segmentation part, we generalize the scale-space continuation method of the gradient-based active contour model to obtain the wavelet-based active contour model. The moduli of wavelet transform coefficients have been introduced into the energy function of the active contour model, and the scale-space continuation process has been applied from coarse to fine scales in the wavelet transform. For the object tracking part, a multi-resolution motion estimation method using the wavelet modulus method and another cycle-spinning-like motion estimation method have been proposed for motion estimation of a snake contour in consecutive frames. The proposed methods improve the ability to track objects with large amounts of motion or deformation. Our framework can be embedded in the wavelet-based video coding system for advanced application.

## References

- [1] ISO/IEC JTC1/SC29/WG11 N3747, MPEG-4 Overview(v.16), Oct. 2000.
- [2] ISO/IEC JTC1/SC29/WG11 N3349. MPEG-7 Overview(v2.0), Mar. 2000.

- [3] H.-H. Wu, J.-C. Liu, and C. Chui, "A wavelet-frame based image force model for active contouring algorithms", *IEEE trans. on Image Processing*, vol.9, no. 11, Nov. 2000.
- [4] H. Yoshida, S. Katsuragawa, Y. Amit, K. Doi, "Wavelet snake for classification of nodules and false positives in digital chest radiographs", *Proc. of 19th Intl. Conf. of the IEEE Engineering in Medicine and Biology Society*, 1997.
- [5] Y.-T. Lin and Y.-L. Chang, "Tracking deformable objects with the active contour model", *IEEE Intl. Conf. on Multimedia Computing and Systems*, pp. 606–609, 1997.
- [6] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models", *Proc. First Intl. Conf. Computer Vision*, pp. 259–268, 1987.
- [7] F. Leymarie and M. D. Levine, "Tracking deformable objects in the plane using an active contour model", *IEEE Trans. on PAMI*, pp 617–634, 1993.
- [8] A. Witkin, D. Terzopoulos, and M. Kass, "Signal matching through scale space", *Proc. 5th Nat. Conf. Artificial Intel.*, pp 714–719, 1986.
- [9] A. Witkin, "Scale-space filtering", *Proc. 8th Intl. Joint Conf. Artificial Intel.*, pp. 1019–1021, 1983.
- [10] S. Zafar, Y.-Q. Zhang, and B. Jabbari, "Multiscale video representation using multiresolution motion compensation and wavelet decomposition", *IEEE J. on Selected Areas in Comm.*, pp 24-32, 1993.
- [11] Y.-Q. Zhang and S. Zafar, "Motion-compensated wavelet transform coding for color video compression", *IEEE Trans. on Circuits and Systems for Video Tech.*, pp 285–296, 1992.
- [12] J. Magarey and N. Kingsbury, "Motion estimation using a complex-valued wavelet transform", *IEEE Trans. on Signal Processing*, pp 1069–1084, Apr. 1998.

- [13] S. Mallat and S. Zhong, “Characterization of signals from multiscale edgels”, *IEEE Trans. on PAMI*, pp 710–732, 1992.
- [14] S. Mallat, “A wavelet-tour of signal processing”, *Academic Press*, 1998.
- [15] R. R. Coifman and D. L. Donoho, “Translation-invariant de-noising”, *Lecture Notes in Statistics: Wavelet and Statistics*, pp. 125–150, 1995.
- [16] A. Murat Tekalp, “Digital video processing”, *Prentice Hall*, 1995.
- [17] M. Bierling, “Displacement estimation by hierarchical block-matching”, *Proc. Visual Comm. and Image Proc.*, SPIE, pp. 942-951, 1988.
- [18] K. Wu, D. Gauthier, and M. D. Levine, “Cell segmentation revisited”, *McRCIM Tech. Rep. CIM-91-4*, McGill University, Canada, 1991.
- [19] J. Lee, A. Ishihara, G. Oxford, B. Johnson, and K. Jacobson, “Regulation of cell movement is mediated by stretch-activated calcium channels”, *Nature*, 400(6742):382-6, 1999.
- [20] W.N. Lian, J.W. Tsai, P.M. Yu, T.W. Wu, S.C. Yang, Y.P. Chau, and C.H. Lin, “Targeting of Aminopeptidase N to Bile Canaliculi Correlates with Secretory Activities of the Developing Canalicular Domain”, *Hepatology*, No. 30, pp. 748-760, 1999.

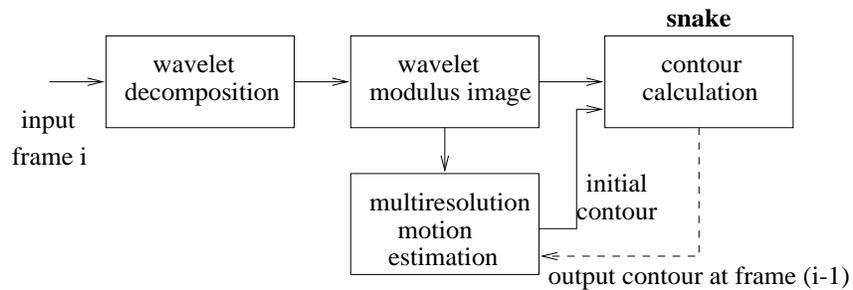


Figure 1: Framework of the proposed wavelet-based active contour model for object segmentation and tracking.

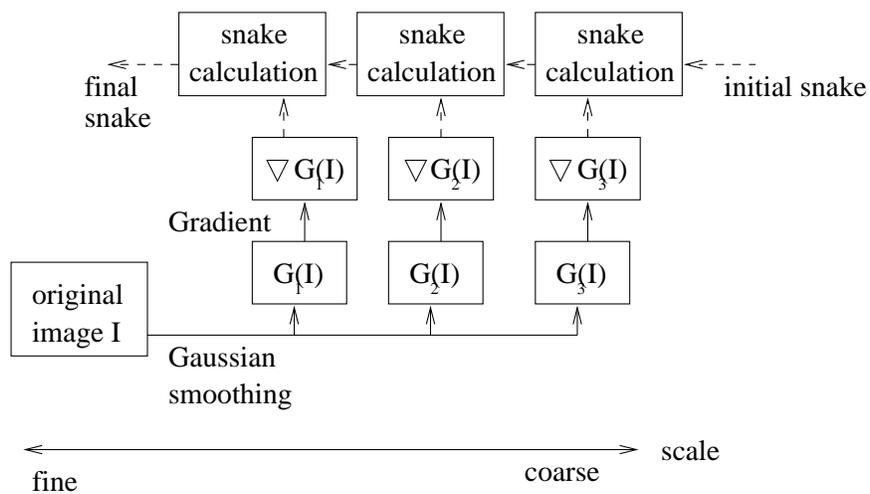
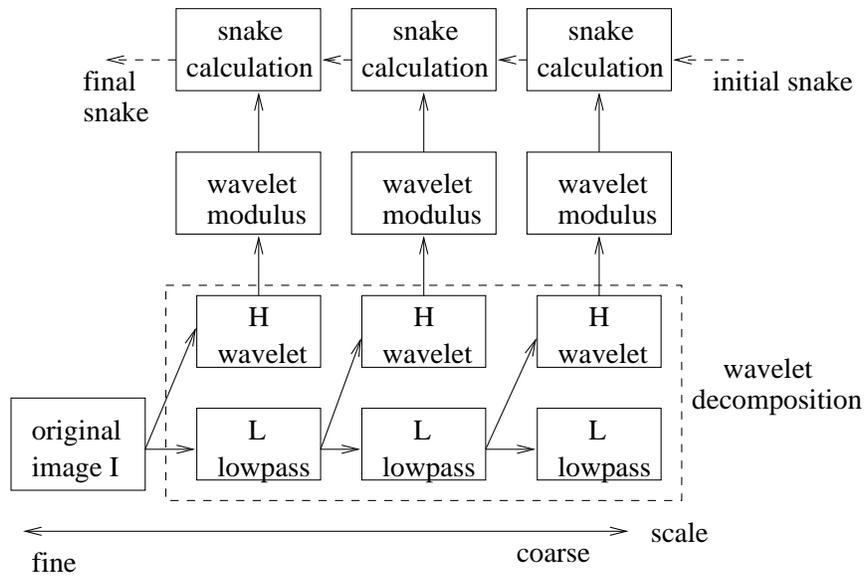
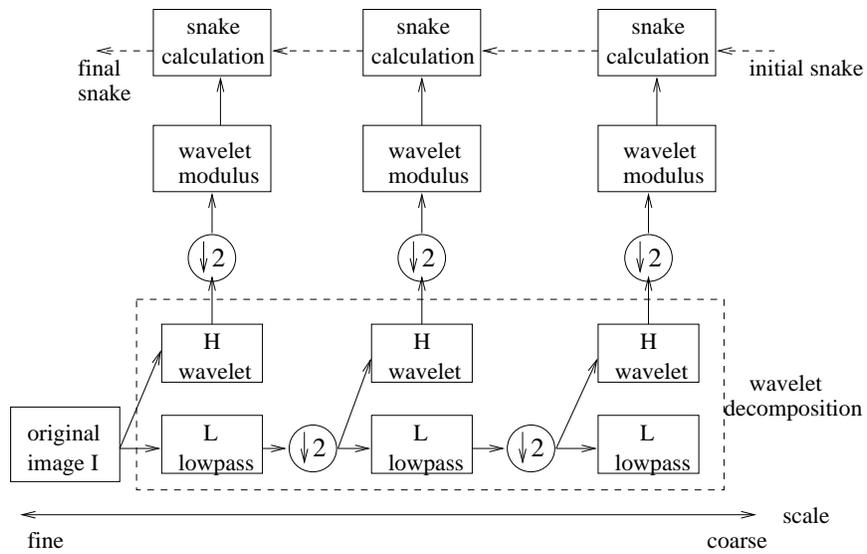


Figure 2: Flow chart of the gradient-based scale-space continuation method for the active contour model.



(a)



(b)

Figure 3: Flow chart of the wavelet-based active contour model for object segmentation.

(a) Without down-sampling, (b) with down-sampling.

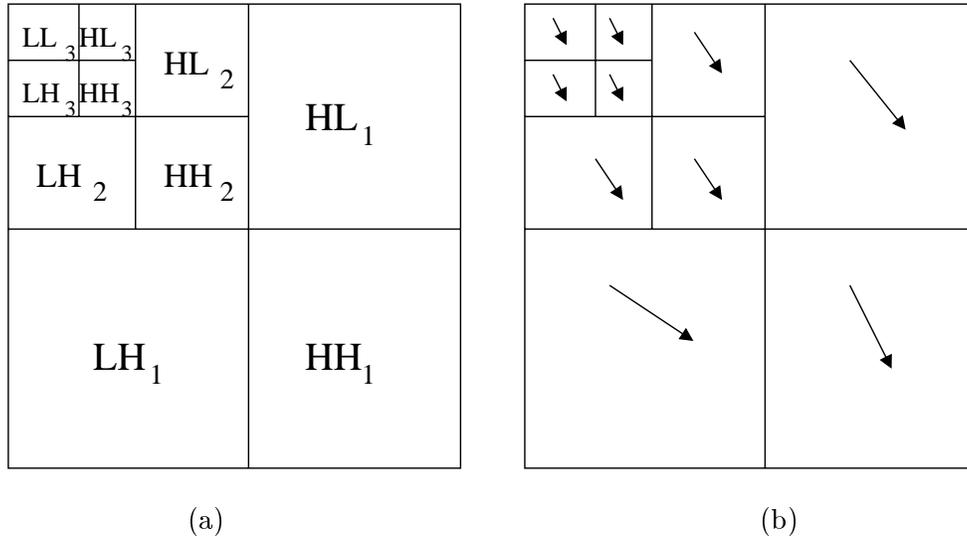


Figure 4: (a) Subband structure of wavelet transform. (b) Multi-resolution motion estimation results of (a).

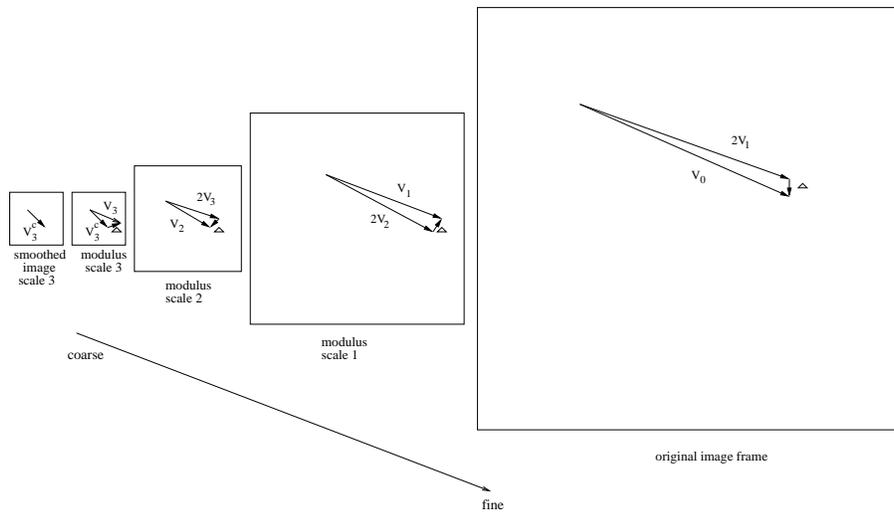


Figure 5: Example of multi-resolution motion estimation using the wavelet modulus.

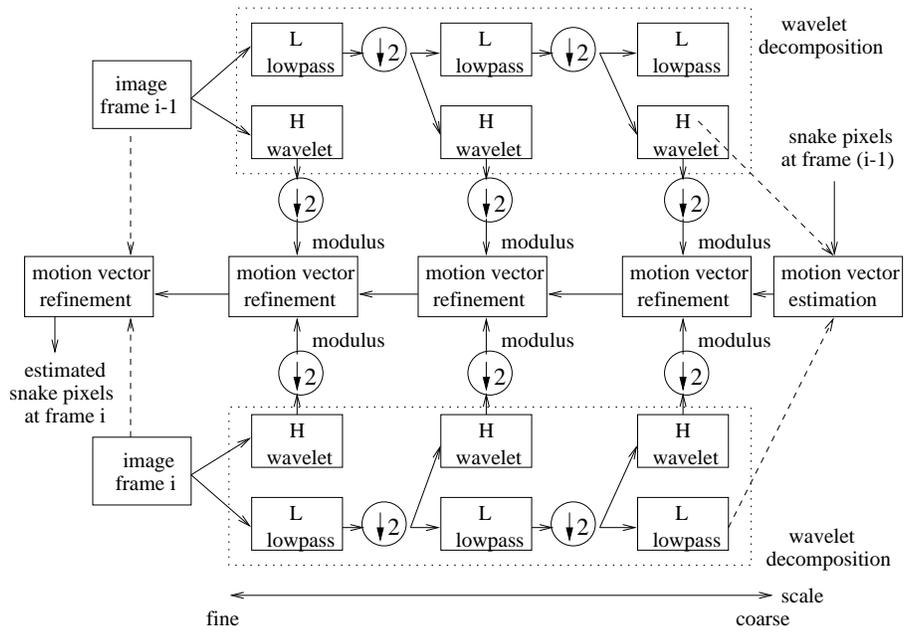


Figure 6: Flow chart of multi-resolution motion estimation using the wavelet modulus.

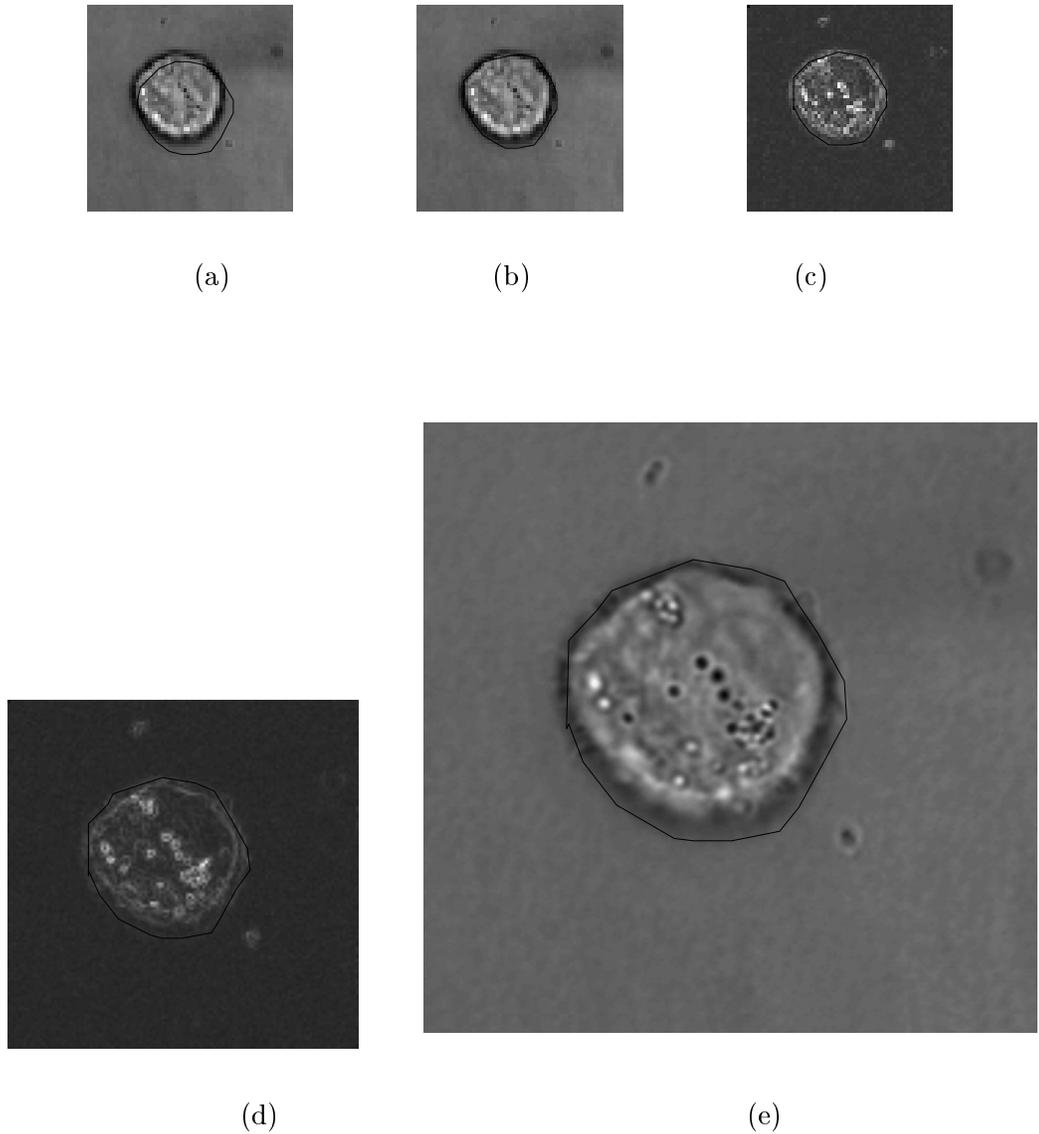
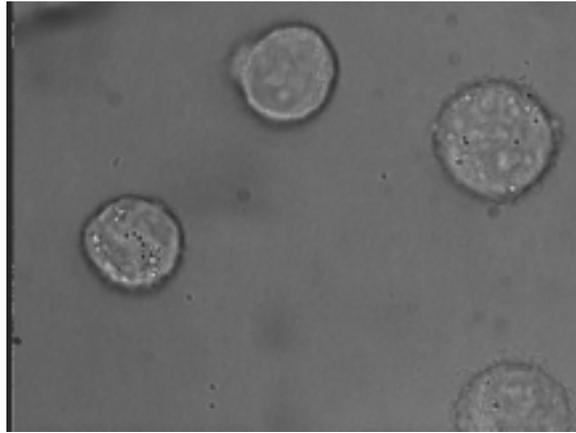
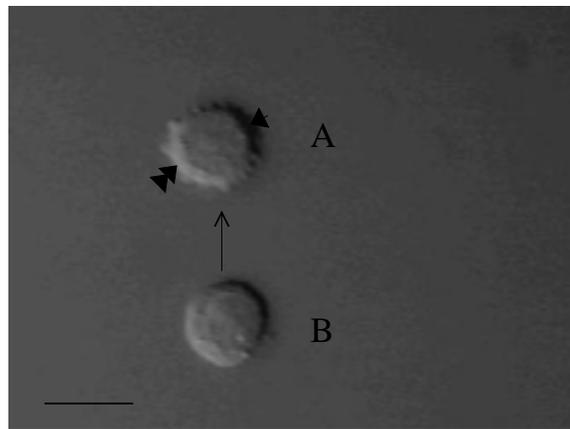


Figure 7: Example of motion estimation.(a) The snake contour of the previous frame is shown at scale  $2^2$ (low frequency) in the current frame. (b)-(e) Motion estimation results are shown at scale  $2^2$ (low frequency), scale  $2^2$ (high frequency modulus), scale  $2^1$ (high frequency modulus), and the scale of the original image.

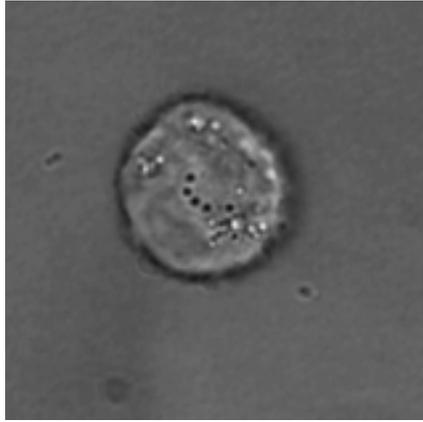


(a)

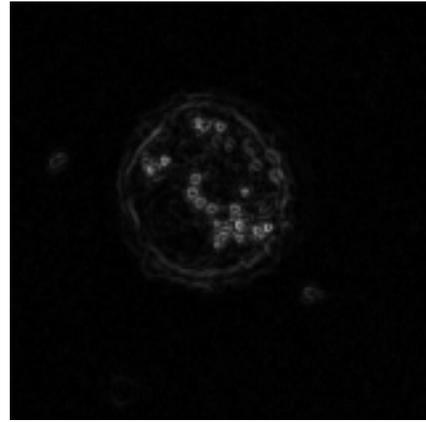


(b)

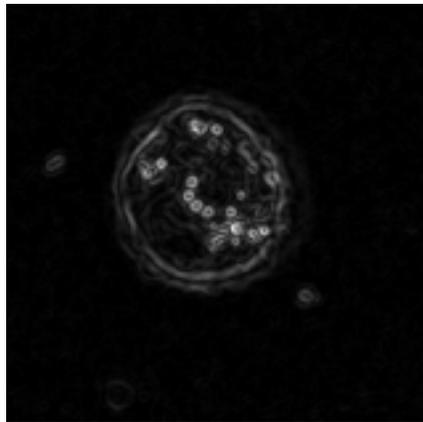
Figure 8: First frame in the test image sequences. (a) Image sequence 1, (b) image sequence 2.



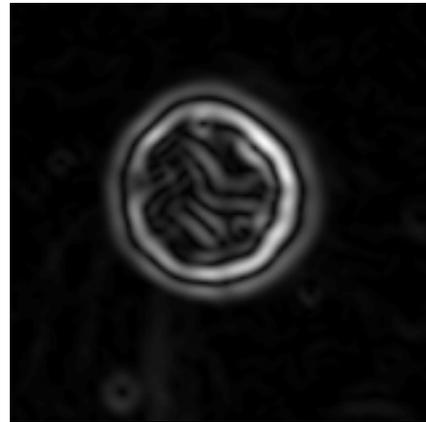
(a) Original image



(b) Scale  $2^2$



(c) Scale  $2^3$



(d) Scale  $2^4$

Figure 9: (a) Original image. (b)-(d) Wavelet transform modulus at scales  $2^2$ ,  $2^3$ , and  $2^4$ .

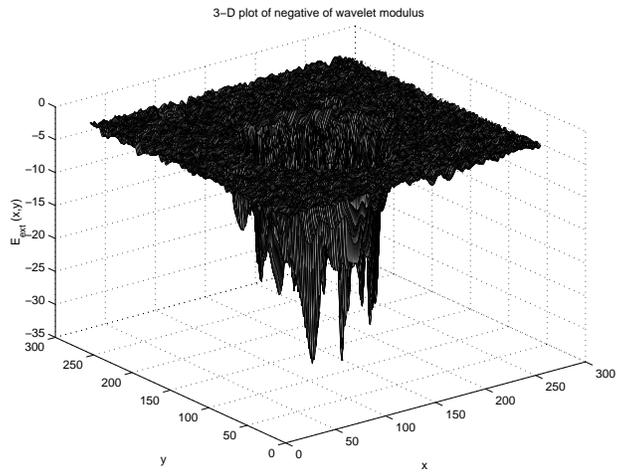
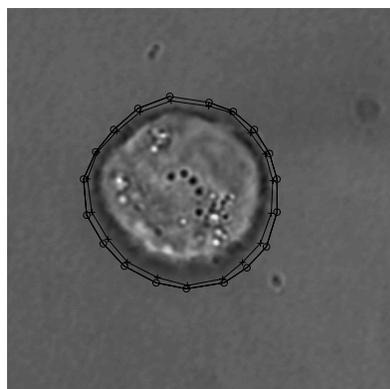
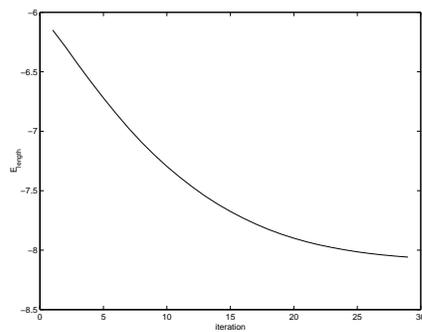


Figure 10: 3-D plot of Fig. 9(c), where the height corresponds to the negative value of the gray-level intensity.



(a)



(b)

Figure 11: (a) 'o':initial contour; '+' :converged snake contour. (b) Plot of  $E_{\text{length}}(v)$ .

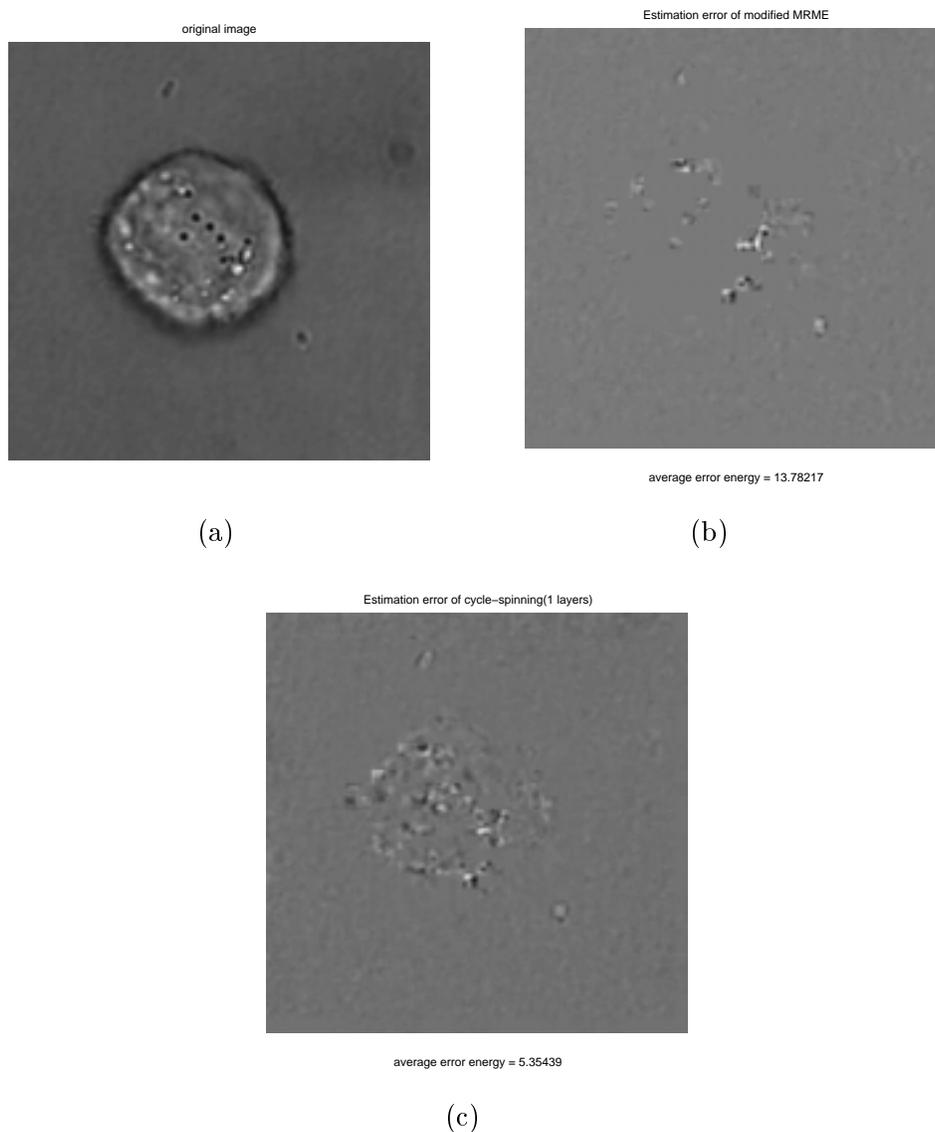
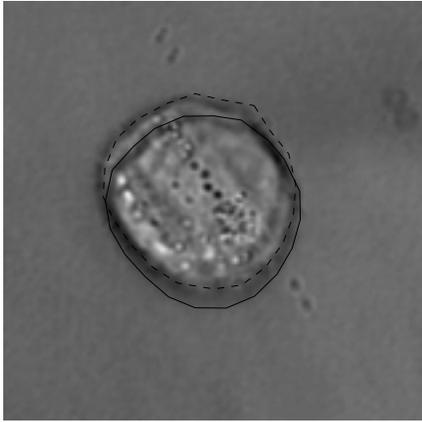
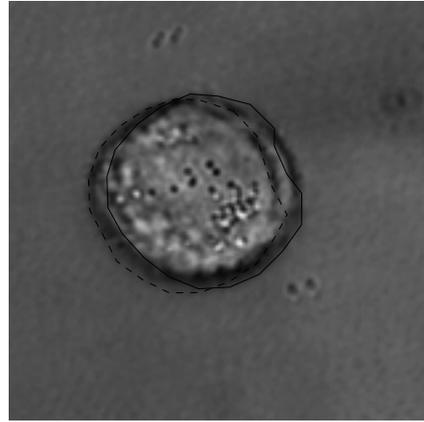


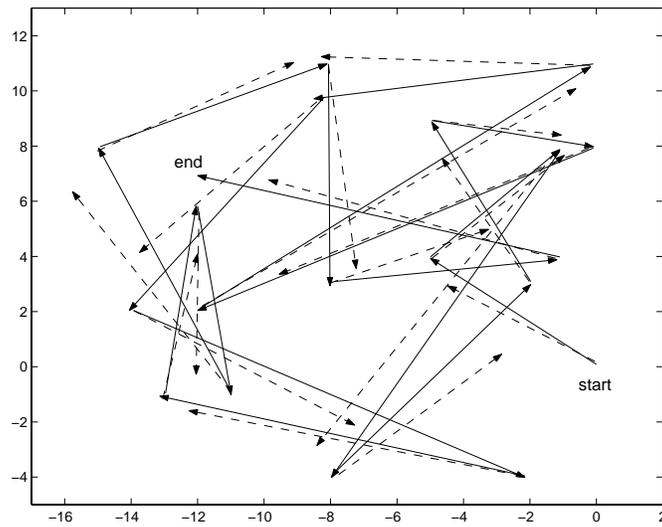
Figure 12: Experiment in motion estimation. The original image in (a) was shifted up and to the left 10 pixels to create a shifted image. (b) and (c) show the differences between the original image and the estimated images which were reconstructed using the motion vectors and the original image. (b) Results using multi-resolution motion estimation using the wavelet modulus, where  $MSE = 13.78$ . (c) Results using cycle-spinning-like motion estimation with a  $3 \times 3$  shift range, where  $MSE = 5.35$ .



(a)

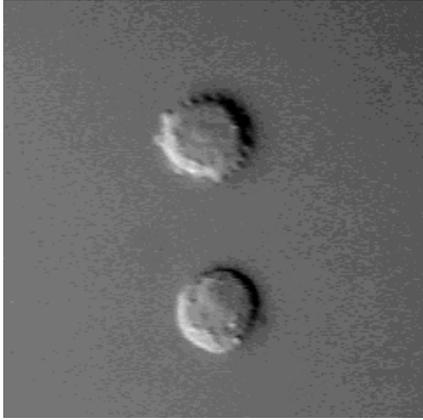


(b)

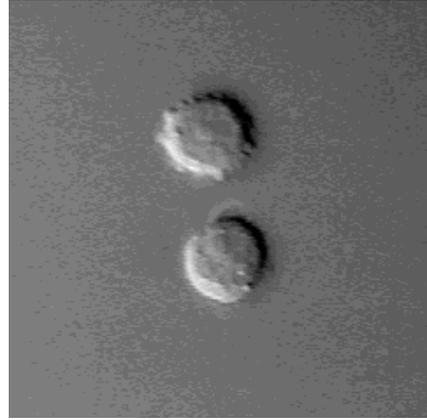


(c)

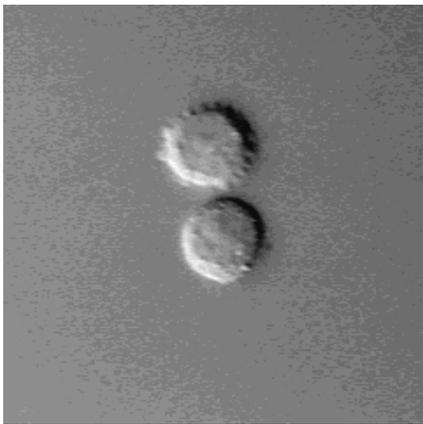
Figure 13: Results of random cell motion. (a) Overlapping of frames 3 and 4 and the extracted contours. (b) Overlapping of frames 12 and 13 and the extracted contours. (c) Solid line: the added random movement. Dashed line: the estimated motion vector obtained using our method.



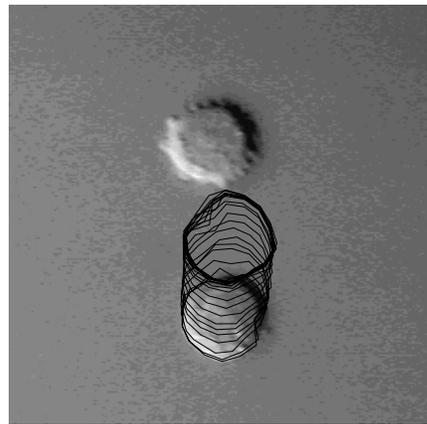
(a)



(b)



(c)



(d)

Figure 14: (a)-(c) Extracted frames. (d) The trace of the cell contour.