World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:8, No:8, 2014

# Particle Filter Supported with the Neural Network for Aircraft Tracking Based on Kernel and Active Contour

Mohammad Izadkhah, Mojtaba Hoseini, Alireza Khalili Tehrani



***Abstract*—**In this paper we presented a new method for tracking flying targets in color video sequences based on contour and kernel. The aim of this work is to overcome the problem of losing target in changing light, large displacement, changing speed, and occlusion. The proposed method is made in three steps, estimate the target location by particle filter, segmentation target region using neural network and find the exact contours by greedy snake algorithm. In the proposed method we have used both region and contour information to create target candidate model and this model is dynamically updated during tracking. To avoid the accumulation of errors when updating, target region given to a perceptron neural network to separate the target from background. Then its output used for exact calculation of size and center of the target. Also it is used as the initial contour for the greedy snake algorithm to find the exact target's edge. The proposed algorithm has been tested on a database which contains a lot of challenges such as high speed and agility of aircrafts, background clutter, occlusions, camera movement, and so on. The experimental results show that the use of neural network increases the accuracy of tracking and segmentation.

***Keywords*—**Video tracking, particle filter, greedy snake, neural network.

## I. Introduction

AIRCRAFT tracking, is one of the major issues addressed in the air control system. Aircraft tracking using video images have many applications such as aircraft guidance during landing, automatic scoring in acrobatics and weapons targeting on military matters. There are some important challenges in tracking that makes it difficult. The existence of similar objects, occlusion, background clutter, changes in luminance, and so on. This paper proposes a new method to deal with these challenges. The rest of the paper is organized as follows. Section II gives a brief survey of tracking algorithms. The proposed method is discussed in Section III. Results are presented in Section IV and Section V concludes the paper.

## II. Related Work

In [1] a variety of tracking methods have been investigated in detail. For ease of discussion, we classify these methods into two categories: region-based method and contour-based method [2]. The basic idea of region-based method is to track object with the similarity measure of object region. mean-shift algorithm has achieved considerable success in similarity region search due to its simplicity and robustness. The scale of the mean-shift kernel is a crucial parameter, so many mechanisms were presented for choosing or updating scale. for example [3] suggests repeating the mean-shift algorithm at each iteration using window size of ±10% of the current size, and evaluating which scale is best using the Bhattacharyya coefficient. But these mechanisms have low efficiency in sudden deformations. Moreover, non-deterministic methods such as the kalman filter [4] and particle filter [5] are more effective for dealing with occlusion. In general, estimation methods have the major advantage over other methods because if they fail in one or more frames to detect a target, they can retrieve themselves in the next frames and continue to tracking. In [6] for the purpose of tracking the aircraft, extracted local features are used. This tracking algorithm lies in the assumption that there is little motion between consecutive frames.

The contour based method, uses dynamic curve model for target tracking. This model at the first time was introduced in 1987 by Kass [7] and includes an initial curve and its energy function. Then in 1988, Amini [8] presented a novel method using dynamic programming and Williams in [9] proposed a greedy algorithm for this model. These three methods are compared with each other in [10]. Both dynamic programming and greedy algorithm are better than the original algorithm. The time complexity of the dynamic programming approach is too large and thus this approach is not suited for tracking a contour in real time without specialized hardware. Nevertheless, this approach might be applied to static image analysis. The greedy algorithm is computationally inexpensive and therefore well suited for real time applications. This algorithm reduces the complexity of the basic method from $O(nm^3)$ to $O(nm)$ where n is number of points and m is the size of the neighborhood. Contour-based methods can achieve a high tracking precision, but in targets with high speed or large aspect change, their robustness is usually not better than that of region-based methods. Because the most methods for contour based tracking, use the contour of the current frame as the initial contour in the next frame. So far, various solutions such as defining a new external energy [11] is presented. [12] Employ game of life cellular automaton to manage snake pixels deformation in each epoch of minimization procedure. Another problem of contour-based methods is high computational cost, especially for large objects, or objects with high speed.

Some tracking methods use both region and contour information. [2] introduced a two stage object tracking

Mohammad Izadkhah, ICT Department, MalekAshtar University of Technology, Tehran, Iran (e-mail: mmlion.izadkhah@yahoo.com).
Mojtaba Hosseini, Computer Engineering Department, Amirkabir University of Technology, Tehran, Iran, (e-mail: mojtabahoseini@aut.ac.ir).
Alireza Khalili Tehrani, ICT Department, Malek Ashtar University of Technology, Tehran, Iran.

method. First, the kernel-based method has been used to locate the object region. In the next stage kalman filter with Bhattacharyya coefficient are used to determine the object tracking position. Finally diffusion snake [13] is used to evolve the object contour.

## III. Proposed Method

Compared to the kalman filter, particle filter for use of random sampling is more robust against issues such as occlusion. In addition, particle filter does not consider any conditions about the model. For this reason, we used particle filter for estimation of target location. In proposed method, both region and contour information has been used to create target model. A major problem in the use of systems that continuously update their model, is the accumulation of the errors after a few update that it cause the tracker to fail.

In our method, target region given to a perceptron neural network to separate the target from background. This makes the location estimation error decreases and so tracking parameters such as size, speed and target model will update correctly. Finally its output will be used as the initial contour of the greedy snake algorithm to find the exact target's edge. Another advantage of using neural network is reducing runtime.

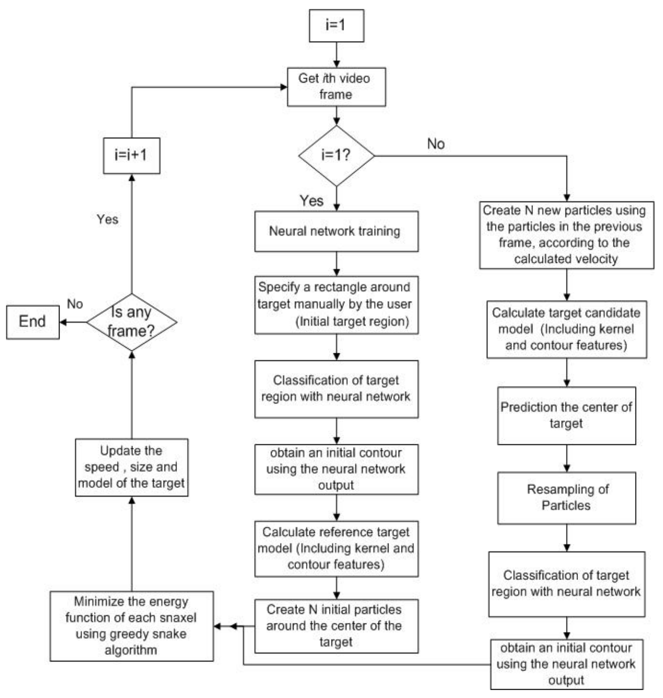



Fig. 1 The flowchart of proposed algorithm. Our proposed method is made in three steps: (1) estimate the target location by particle filter, (2) segmentation target region using neural network, and (3) find the exact contours by greedy snake algorithm

### *A. Estimate Target Location*

Estimation methods are based on the uncertainty in the corresponding objects in successive frames. This view is derived from this fact that receiving means and sensors in imaging have noise and also behavior of an object is comply with random pattern. Statistical methods control the problems that arising from measurement errors by using uncertainties.

Two most important non-deterministic tracking methods, are Kalman filter and particle filter. Below some advantages of particle filter compared to the kalman filter are discussed.

- Kalman filtering assumes white noise with a Gaussian probability distribution. but the particle filter is not limited to this assumption.
- Kalman filter considers that the model of system is linear. but particle filter can used for both linear and non-linear models.
- Particle filters expand their possibilities on improbable places and this property helps it to find targets with high speed and sudden direction changes.
- For these reasons, we used particle filter for estimation of target location.

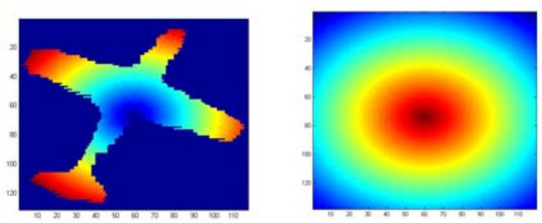
### *B. Target Representation*

In fact Object representation is extraction of features that are useful for target tracking. In this work combination of these features was used for target representation: variance and mean of each bound in RGB space, probability density function of silhouette target region and probability density function of rectangular target region (target and background) in HSI space.

Color histogram is calculated by using a weighting scheme. Different pixel’s contribution to the object representation depends on its position with respect to the center of the target. Greate weights are given to pixels near the region center and smaller weights are assigned to pixels farther from the center. Using these weights increases the robustness of the density estimation since the peripheral pixels are the least reliable, being often affected by occlusions (clutter) or interference from the background. We use Epanechnikov kernel k(x) as a weighting function.

$$k(x) = \begin{cases} \frac{1}{2} c_d^{-1} (d+2)(1-\|x\|^2), if \ \|x\|^2 \le 1; \\ 0, otherwise, \end{cases} \quad (1)$$

where $c_d$ is the volume of the unit $d$-dimensional sphere. Fig. 2 shows an example of function k(x) that is calculated in both silhouette and rectangular region.



(a) (b)

Fig. 2 The output of Epanechnikov kernel for (a) rectangular target region (b) silhouette target region

To reduce the computational cost, m-bin histograms are used. The function b: $R^2 \rightarrow \{1...m\}$ associates to the pixel at location the index b( ) of its bin in the quantized feature space. The HSI space for proposed method is quantized into 12 bins. The probability of the feature u=1…m in the target model is then computed as:

$$\hat{q}_u = C\sum_{i=1}^{n} k(\|x_i^*\|^2)\delta[\mathrm{b}(\mathrm{x}_i) - u] \tag{2}$$

where $\delta$ is the Kronecker delta function. The normalization constant is derived by imposing the condition $\sum_{u=1}^{m} \hat{q}_u = 1$, from where:

$$C = \frac{1}{\sum_{i=1}^{n} k(\|x_i^*\|^2)} \tag{3}$$

since the summation of delta functions for u=1...m is equal to one.

Let $\{X_i\}_{i=1....nh}$ be the normalized pixel locations of the target candidate, centered at $y$ in the current frame. The tracking algorithm searches for the target in this frame, from the target candidates. The probability density function for target candidates are represented by:

$$\hat{p}_u(y) = C_h \sum_{i=1}^{n_h} k\left(\left\|\frac{y - x_i}{h}\right\|^2\right)\delta[\mathrm{b}(\mathrm{x}_i) - \mathrm{u}] \tag{4}$$

where $h$ is the bandwidth, and $C_h$ is the normalization constant and is defined as:

$$\mathrm{C}_h = \frac{1}{\sum_{i=1}^{n_h} k\left(\left\|\frac{y - x_i}{h}\right\|^2\right)} \tag{5}$$

We adopt the Bhattacharyya coefficient to calculate the distance among target pdf and candidates.

$$\hat{\rho}(y) = \rho[\hat{p}(y), \hat{q}] = \sum_{u=1}^{m} \sqrt{\hat{p}_u(y)\hat{q}_u} \tag{6}$$

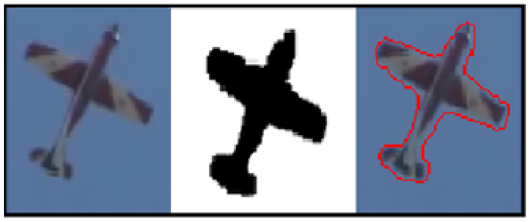
The similarity function defines a distance among target model and candidates. We are assumed that the distribution of similarity function $F$ is Gaussian. If $V_i$ and $M_i$ denotes the variance and mean difference between two models in $i$th bound of RGB space, respectively, then the Gaussian measurement likelihood function is computed as

$$F = \mathrm{e}^{\frac{1}{2\sigma^2}\sum_{i=1}^{3}(V_i + M_i + \hat{\rho}_i)} \tag{7}$$

where $\hat{\rho}_i$ calculate with eq.6 in HIS space and σ is the standard deviation of the distribution of distance measures.

*C. Segmentation Target Region*

Most methods for contour based tracking use the contour of the current frame as the initial contour in the next frame. This approach in tracking targets with high speed or large aspect change, will fail. In the proposed method the neural network is used to obtain an initial contour. For this purpose, the target region which is estimated in the previous stage, segmented into two categories, target and background by the neural network and then extract the initial contour from this binary image. For this purpose, we have used a three-layer perceptron neural network. Test results show that use of neural network to obtain the initial contour, increases the speed and accuracy of the greedy snake algorithm and also there is not the problem of not converge to the concave parts of the target. These advantages are due to the initial contour is very close to the final contour.



(a) (b) (c)

Fig. 3 An example of the implementation of the segmentation stage (a) An aerobatics aircraft (b) Output of neural network (c) The initial contour obtained by the segmentation stage

*D. Greedy Snake*

A discrete snake is defined as $N$ contour points named snaxels and represented as: $V=\{v_i=[x_i,y_i] \mid 1<=i<=N\}$ where for a closed snake v1=vn+1. In greedy algorithm for minimization of snake energy, a neighborhood is defined for each snaxel of snake and snake energy is computed for each point of neighborhood. The snaxel (center of neighborhood) is then updated to a point of neighborhood that has minimum energy. For locking of snake on target boundary this process should be iterated several times. The snaxel energy consists of two parts including internal and external energies:

$$E_{contour} = \sum_{n=1}^{N} E_{\mathrm{internal}}(v_n) + \sum_{n=1}^{N} E_{external}(v_n) \tag{8}$$

The internal energy function defines shape characteristics of the contour and is defined using the following:

$$E_{\mathrm{internal}}(v_n) = \alpha E_{continuity}(v_n) + \beta E_{curvature}(v_n) \tag{9}$$

where $E_{continuity}(v_n)$ and $E_{curvature}(v_n)$ are the normalized continuity and curvature energies, for the snaxel $Vn$, respectively. $E_{continuity}$ controls the distance between points. The higher α, means that minimize the distance between points is more important. $E_{curvature}$ controls the smoothness of the curvature. The higher β, means that maximize the angles between points is more important.

The continuity and curvature energy functions are defined as:

$$E_{continuity} = \frac{d - |v_{n(j)} - v_{n-1}|}{\max\{d - |v_{n(j)} - v_{n-1}|\}} \tag{10}$$

$$E_{curvature} = \frac{|v_{n-1} - 2v_n + v_{n+1}|^2}{\max\{|v_{n-1} - 2v_n(j) + v_{n+1}|^2\}} \tag{11}$$

where $d$ is the average curve length and $V_{n(j)}$ represents the neighbors of a point $V_n$ for $j$=1,2,…,$M$. $M$ is number of pixel in neighborhood.

External energy defines the using of image data in the process of minimization. The External energy is defined as

$$E_{external} = \frac{\min(mag) - mag}{\max(mag) - \min(mag)} \tag{12}$$

where *mag* is the gradient magnitude of the current point, *max* is the maximum and *min* is the minimum gradient in each neighborhood. This gradient magnitude term is negative so that points with large gradient will have small values. Fig. 4 shows output of the greedy snake algorithm on a sample frame.

Fig. 4 One example of the adaptation of the initial contour to the edges of the target. Red contour obtained by the neural network, and white contour has been created the greedy snake stage

## IV. Experimental Results

The proposed method was implemented with matlab program. The video database was downloaded from [14]. This database was collected during the Air Race 2006 in Perth, Australia. Most of the data was collected in AVI format at 25 frames per second and a frame size of 720×576. There are 27 video sequences of varying length. The longest video sequence lasts for over one minute. In addition to these 27 sequences, there are four video sequences acquired with a low resolution camera at 15 frames per second and a frame size of 640×480. The database contains six different types of aircrafts including propeller planes, jet planes, aerobatics planes, fighter, transport aircraft and a helicopter. Some of the challenges, from tracking point of view, in the database include: camera movement and optical zoom, high speed aircrafts, aircraft pose variation, occlusions due to clouds and trees, background clutter, and smoke from aerobatics aircrafts.

In this section some results of the proposed algorithm is presented. Fig. 5 shows the tracking results of three video sequences, aerobatic, transport, and jet aircraft, with proposed method. Fig. 6 indicates that our method can cope with the background clutter and also partial and complete occlusions caused by clouds.

In order to evaluate the tracking precision quantitatively, we have used the tracking error for a video frames, as defined

$$TE(t) = 1 - \frac{F_1 \cap F_2}{F_1} \tag{13}$$

where $t$ is frame number (time) and $F_1$ and $F_2$ are the set of pixels within target contour in the manual segmentation result and segmentation result with the object tracking method, respectively. Fig. 7 shows the tracking error for three videos that are shown in Fig. 5.

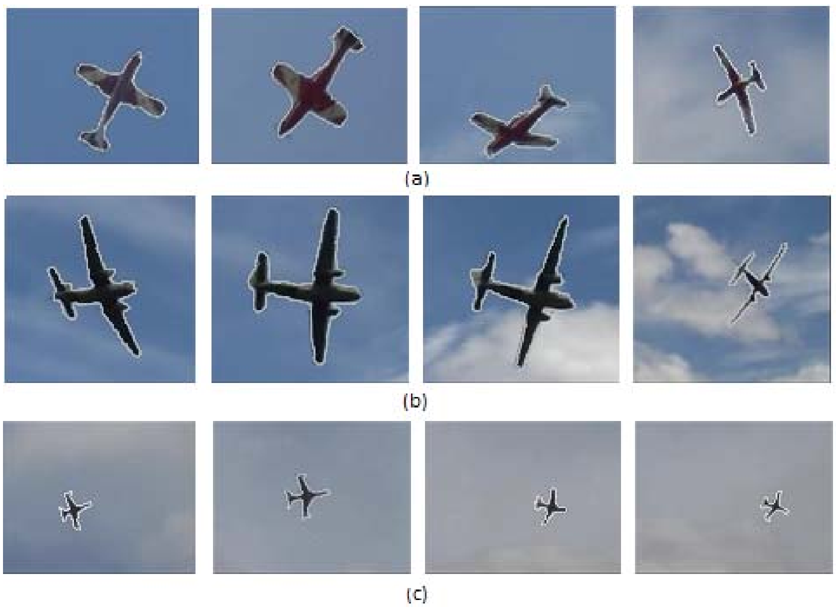



Fig. 5 Tracking of the three video sequences (a) aerobatics aircraft (b) transport aircraft (c) jet aircraft

Fig. 6 Tracking in the presence of occlusions due to clouds

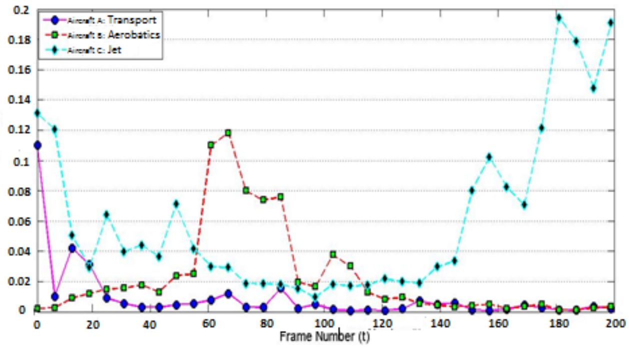



Fig. 7 tracking error for three video sequences shown in Fig. 5

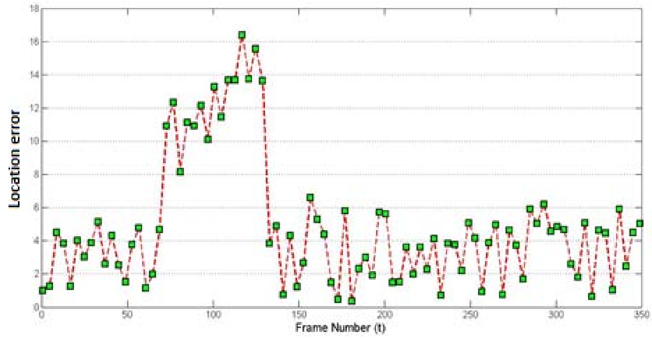



Fig. 8 Location error for video sequence that shown in Fig. 5 (a)

The location error of object center is adopted to evaluate the location precision, which is defined as follows:

$$d_i = \| C_a - C_m \| \tag{14}$$

where $C_a$ and $C_m$ are the object center coordinates of the target tracking and manual segmentation results, respectively. $d_i$ is Euclidean distance between two center in frame $i$. Fig. 8 shows the location error during target tracking.

To show the effect of neural network on tracking, segmentation stage was removed from the proposed algorithm. The results of these tests are given in Table I. Speed and accuracy of particle filter, largely depend on the number of particles. These results show that contribute to the particle filter by neural network, due to its high power in detecting targets, while increases the accuracy of tracking, reduces the number of particles and thus reduces the computation time.

TABLE I
EFFECT OF NEURAL NETWORK ON TRACKING

| video | Number of total frames | Number of particles | Without using neural network: Number of frames tracked until the first loss | Without using neural network: Mean Error (pixel/frame) | Without using neural network: Average processing time (sec/frame) | Using neural network: Number of particles | Using neural network: Number of frames tracked until the first loss | Using neural network: Mean Error (pixel/frame) | Using neural network: Average processing time (sec/frame) |
|---|---|---|---|---|---|---|---|---|---|
| Multiple smoke | 285 | 100 | 180 | 9.8 | .2 | 50 | 252 | 1.54 | .17 |
| Aerobatics | 415 | 100 | 71 | 10.33 | .97 | 50 | 415 | 2.68 | .41 |
| Jet | 647 | 100 | 140 | 6.48 | .59 | 50 | 647 | 2.32 | .22 |
| Jet in clouds | 271 | 100 | 13 | 1.3 | .15 | 50 | 84 | .153 | .11 |

## V. CONCLUSION

In this paper we are presented a new method for tracking flying targets. The proposed method is made in three steps, estimate the target location by particle filter, segmentation target region using neural network and find the exact contours by greedy snake algorithm. In the proposed method we have used both region and contour information to create target candidate model and this model dynamically updated during tracking. The proposed algorithm has been tested on a database which contains a lot of challenges such as high speed and agility of aircrafts, background clutter, occlusions, camera movement and so on. The experimental results show that the use of neural networks has two major advantages. 1) Enhance tracking precision due to its ability to detect objects, and thus avoiding the accumulation of errors during tracking and 2) enhance the accuracy of segmentation target in greedy snake, because of accurate initial contour. Also the experimental results show that the proposed algorithm compared to the algorithm that does not use the neural network has a higher speed. Because the neural network reduces the number of particles in particle filters and therefore calculations per frame are reduced. But compared to some other methods, such as [6] our method is time consuming. Also in the proposed method, the extracted features are only based on color, and for this reason our method cannot effectively track the object when the color feature of the object is very similar to that of the background.

## REFERENCES

[1] A. Yilmaz, O. Javed, and M. Shah, "Object tracking," *ACM Computing Surveys,* vol. 38, no. 4, p. 13–es, Dec. 2006.

[2] Q. Chen, Q. Sun, P. A. Heng, S. Member, and D. Xia, "Two-Stage Object Tracking Method Based on Kernel and Active Contour*," Circuits and Systems for Video Technology*, vol. 20, no. 4, pp. 605–609, 2010.

[3] D. Comaniciu and V. Ramesh, "Kernel-Based Object Tracking," *Pattern Analysis and Machine Intelligence*, vol. 25, no. 5, pp. 564–577, 2003.

[4] P.Maybeck, "Object tracking using affine structure for point correspondences," *Computer vision and pattern recognition*, 1997, pp. 704–709.

[5] M. Isard and A. Blake, "Condensation — Conditional Density Propagation for Visual Tracking," *computer vision*, vol. 29, no. 1, pp. 5–28, 1998.

[6] A. S. Mian, "Realtime Visual Tracking of Aircrafts," *Digital Image Computing: Techniques and Applications,* pp. 351–356, 2008.

[7] D. terzopoulo. M.Kass, A.Witkin, "snakes active countour models," *computer vision*, pp. 321–331, 1988.

[8] A. Amini, S. Tehrani, and T. E. Weymouth, "using dynamic programming for minimizing the energy of active contours in the presence of hard constraints," *Pattern Analysis and Machine Intelligence,* pp. 855–867, 1988.

[9] M. S. J.Williams, "A fast algorithm for active contour (greedy snake)," *Image understanding*, vol. 55, pp. 14–26, 1992.

[10] J. Denzler and H. Niemann, "Evaluating the performance of active contour models for real time object tracking," *Asian Conference on Computer Vision*, vol. 2, no. Informatik 5, 1995.

[11] C. Xu, S. Member, J. L. Prince, and S. Member, "Snakes , Shapes , and Gradient Vector Flow," vol. 7, no. 3, pp. 359–369, 1998.

[12] S. Sabouri, A. Behrad, and H. Ghassemian, "Deformable Contour-Based Maneuvering Flying Vehicle Tracking in Color Video Sequences," *ISRN Machine Vision*, vol. 2013, pp. 1–15, 2013.

[13] D. Cremers, F. T. Auser, and J. Weickert, "Diffusion Snakes : Introducing Statistical Shape Knowledge into the Mumford-Shah Functional," *computer vision,* vol. 50, no. 3, pp. 295–313, 2002.

[14] A. Mian. Home page. http://www.csse.uwa.edu.au/ ajmal