Edge segmentation by Alternating Vector Field Convolution Snakes

Christian Rogaß[†], Sibylle Itzerott[†], Bernd Uwe Schneider[††],
Hermann Kaufmann[†] and Reinhard F. Hüttl[††]

[†] Remote Sensing Section, Helmholtz Centre Potsdam
GFZ German Research Centre for Geosciences, Telegrafenberg, D-14473 Potsdam, Germany
[††] Scientific Executive Board, Helmholtz Centre Potsdam
GFZ German Research Centre for Geosciences, Telegrafenberg, D-14473 Potsdam, Germany

Summary

Active contours or snakes represent widely used methods in edge segmentation. Different, closed snake approaches have been proposed in the past but commonly lack of adequate capture ability and quality, noise robustness and convergence speed. In this paper a new closed approach namely "Alternating Vector Field Convolution (AVFC)" as well as a new force called "Yukawa Alternating Vector Field Convolution (YAVFC)" are proposed which minimize specific snake problems and enhance convergence speed. The comparison of both Gradient Vector Flow and Vector Field Convolution snakes with AVFC and YAVFC snakes determines the advantages of the new closed approaches.

Key words:
Active contour, Snake, External force, Yukawa, Alternating vector field convolution

1. Introduction

The segmentation of images by the extraction of edges can be considered as one of the main tasks in image processing. Assuming an image as a representation of its intensity function local extrema are generally searched to delineate differing elements. Many different methods of edge segmentation are proposed in literature. Active contours also known as snakes belong to modern methods in edge segmentation. In the case of parametric active contours specific parametric curves iteratively move through the image domain until certain mathematic conditions are fulfilled. The most basic condition is the minimization of a specific energy functional which finally contains two terms – the internal energy term and the external energy term. The internal energy term corresponds to the curve itself and controls the elasticity and the rigidity of the curve. The external energy term corresponds to the considered domain and is usually constructed as the potential of the domain. The solution of the minimization problem of the specific energy functional results in a force balance equation, where internal forces representing the internal term are at equilibrium with external forces representing the external energy term. Typical approaches to model the external force are given by Balloon Forces, Gradient Vector Flow and Vector Field Convolution. These approaches address more or less the usual problems of snakes to capture boundary concavities, to overcome remaining image noise and to increase the convergence speed. In most cases they depend on nearby localization of initial contours towards the regions of interest, on a complex and case related determination of snake parameters and the amount of time given for the iterations. These properties do not distort the segmentation process unless similar images are used. For rapid segmentation and interpretation of different samples an extended approach based on the adaption of Vector Field Convolution and Balloon Snakes, Multi Resolution Analysis and alternating snake terms is proposed in this paper. They are called Alternating Vector Field Convolution Snakes (AVFC) respecting the underlying idea of Vector Field Convolution. Additionally a completely new force called Strong Force or Yukawa Force is implemented to cope with prior given general requirements of image segmentation. The related snake is called Yukawa Alternating Vector Field Convolution (YAVFC) snake. Both snake approaches proposed here are evidently able to enhance capture quality by simultaneously increased convergence speed.

2. Materials and methods

2.1. Traditional Snakes

In [1,2,3] an active contour or snake is represented in two dimensions by a planar parametric curve $\underline{v}(s) = [x(s),y(s)]^T$, $s \in [0,1]$ which moves through the spatial domain of an image to minimize the energy functional

$$E = \int_0^1 \frac{1}{2}\left(\alpha \left|\frac{\partial \underline{v}}{\partial s}\right|^2 + \beta \left|\frac{\partial^2 \underline{v}}{\partial s^2}\right|^2\right) + E_{ext}(\underline{v}(s))\, ds \qquad (1)$$

where α and β are weights to control the smoothness and the rigidity of the snake. $\frac{\partial \underline{v}}{\partial s}$ and $\frac{\partial^2 \underline{v}}{\partial s^2}$ refer to the first and second derivative of $\underline{v}$ with respect to s.

E_{ext} denotes the external energy function and is directly derived from the image.

This function becomes small at the regions of interest (ROI), for instance at edges. Taken $I(x,y)$ as the intensity function of an image typical external energy functions are

$$E_{ext}^{(1)}(x, y) = -|\nabla I(x,y)|^2 \qquad (2)$$
$$E_{ext}^{(2)}(x, y) = -|\nabla(G_\sigma(x,y) \otimes I(x,y))|^2 \qquad (3)$$

where $G_\sigma(x,y)$ is the 2D Gaussian function with standard deviation σ, ∇ denotes the gradient or Nabla operator and $\otimes$ represents the linear convolution operator [3]. Minimizing E_{ext} the snake must satisfy the Euler equation

$$\alpha \frac{\partial^2 \mathbf{v}}{\partial s^2} + \beta \frac{\partial^4 \mathbf{v}}{\partial s^4} - \nabla E_{ext}(\mathbf{v}) = 0 \qquad (4)$$

which is typically considered as a force balance equation

$$\mathbf{F}_{int}(\mathbf{v}) = -\mathbf{F}_{ext}(\mathbf{v}) \qquad (5)$$

where the internal force $\mathbf{F}_{int}(\mathbf{v})$ (5) represents $\alpha \frac{\partial^2 \mathbf{v}}{\partial s^2} + \beta \frac{\partial^4 \mathbf{v}}{\partial s^4}$ (4) and controls the smoothness and the rigidity of the snake. There, the smoothness parameter α acts like an elastic membrane and the rigidity parameter β as a thin plate [4]. The external force $\mathbf{F}_{ext}(\mathbf{v})$ (5) deforms the snake towards the ROI. To solve equation (4), the snake is considered dynamically by treating $\mathbf{v}(s)$ as a function of time $t \to \mathbf{v}(s, t)$.

In [3] a solution is obtained when solving the gradient descent equation

$$\alpha \frac{\partial^2 \mathbf{v}}{\partial s^2} + \beta \frac{\partial^4 \mathbf{v}}{\partial s^4} + \mathbf{F}_{ext}(\mathbf{v}) = \frac{\partial \mathbf{v}}{\partial t} \qquad (6)$$

starting from the initial contour $\mathbf{v}(s, t = 0)$.

A numerical solution to (6) can be achieved by using discrete s in a finite difference approach [3]. If the derivatives are approximated with finite differences and $\mathbf{F}_{ext}(\mathbf{v}(s, t))$ expressed as $[\mathbf{F}_x(\mathbf{v}(s,t)), \mathbf{F}_y(\mathbf{v}(s,t))]^T$, the Euler equations can then be written for all snake elements as

$$\mathbf{Ax} + \mathbf{F}_x(\mathbf{v}) = 0 \qquad (7)$$
$$\mathbf{Ay} + \mathbf{F}_y(\mathbf{v}) = 0 \qquad (8)$$

where A is a pentadiagonal banded Matrix.

Assuming $\mathbf{F}_{ext}(\mathbf{v}(s, t))$ constant at each iteration and time step and defining an iterative step size γ the resulting Euler equation is given by

$$\mathbf{Av} + \mathbf{F}_{ext}(\mathbf{v}) = -\gamma \frac{\partial \mathbf{v}}{\partial t} \qquad (9)$$

If $\frac{\partial \mathbf{v}}{\partial t}$ is discretised to $\frac{\partial \mathbf{v}}{\partial t} = \mathbf{v}(s,t) - \mathbf{v}(s, t-1) = \mathbf{v}_t - \mathbf{v}_{t-1}$, then the snake elements or snaxels can be given by the dynamic equation

$$\mathbf{v}_t = \left(\mathbf{A} + \gamma\mathbf{I}\right)^{-1}\left(\gamma\mathbf{v}_{t-1} - \mathbf{F}_{ext}(\mathbf{v}_{t-1})\right) \qquad (10)$$

The parameter γ can also be considered as viscosity parameter and makes the snake harder to deform.

Other approaches to solve the Euler equations are for instance the Finite-Element-Method [4] and the dynamic programming method [5].

2.2. Balloon Snakes

According to [6] the closed, traditional snake has a low capture range if edges are not close enough to the initial contour and the snake tends to shrink in itself if the contour is not subjected to any counterbalancing forces. This property is strongly related to the choice of the parameter γ which directly influences capture range and speed. The balloon model increases the capture capability and speed by normalising the external force and extending it to

$$\mathbf{F}_{ext}(\mathbf{v}) = k_1 \, \mathbf{n}(\mathbf{v}) - k \frac{\mathbf{F}_{ext}(\mathbf{v})}{|\mathbf{F}_{ext}(\mathbf{v})|} \qquad (11)$$

where $\mathbf{n}$ is the unit vector perpendicular to the curve, k_1 is the amplitude of the force and k is the substituted time step [6]. The first term of (11) inflates or deflates the snake like a balloon depending on the choice of k_1. The second term of (11) provides constant curve evolving speed depending on the choice of k. According to [1] closed balloon snakes may not move into boundary concavities depending on the set of k_1 if they are inflated or deflated in the wrong direction as only the magnitudes of the forces are changed and not the directions.

2.3. Gradient Vector Field Snakes

To enhance capture capability for distant edges and boundary concavities, [7] proposes a new external force formulation called Gradient Vector Flow (GVF) field and the respective snake GVF-snake.

The external force $\mathbf{F}_{ext}(\mathbf{v})$ is substituted by defining GVF field as a field $\mathbf{v}(x,y) = [u(x,y),v(x,y)]$ that minimizes the energy functional

$$\varepsilon = \iint \mu \left(\left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial y}\right)^2 + \left(\frac{\partial v}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial y}\right)^2\right) + \nabla f2\mathbf{v} - \nabla f2 dxdy \qquad (12)$$

where ∇f is the gradient of an edge map f, μ is a noise related regularisation parameter and $\frac{\partial u}{\partial x}^2 + \frac{\partial u}{\partial y}^2 + \frac{\partial v}{\partial x}^2 + \frac{\partial v}{\partial y}^2$ is the squared magnitude of the gradient of the optical flow velocity according to [8]. If $|\nabla f|$ gets smaller in nearly homogenous regions or distant from edges, the optical flow term dominates the integrand yielding a smooth field and vice versa.

The GVF field can be found by solving the Euler equations

$$\mu \Delta \mathbf{v} - |\nabla f|^2 |\mathbf{v} - \nabla f| = 0 \qquad (13)$$

where Δ denotes the Laplace operator with $\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$ in the 2D case. Equations (12) and (13) can be solved by treating u and v as functions of time and solving the generalized diffusion equation

$$\mu \Delta \mathbf{v} - |\nabla f|^2 |\mathbf{v} - \nabla f| = 0 = \frac{\partial \mathbf{v}}{\partial t} \qquad (14)$$

With regard to [9] forcing the snake into long indentations is still difficult. This is caused by excessive smoothing of the fields near the boundaries with respect to μ. As an extended approach for GVF-snakes to overcome the indentations problem the Generalized Gradient Vector Flow (GGVF) were proposed where both μ and $|\nabla f|^2$ were substituted to g($|\nabla f|$) and h($|\nabla f|$) in the following equations

$$g(|\nabla f|) \Delta \underline{v} - h(|\nabla f|) |\underline{v} - \nabla f| = \frac{\partial \underline{v}}{\partial t} \qquad (15)$$

$$g(|\nabla f|) = e^{\frac{-|\nabla f|}{K}} \qquad (16)$$
$$h(|\nabla f|) = 1 - g(|\nabla f|) \qquad (17)$$

where K determines the degree of trade-off between field smoothness and gradient conformity [9]. $g(\cdot)$ is a weighting function for a smoothly varying vector field. $h(\cdot)$ is a weighting function which balances the data related $|\nabla f|$.

2.4. Vector Field Convolution Snakes

In accordance to [2,9] the GGVF-snake improves the ability to capture narrow boundary concavities, but is still sensitive to its parameters and to impulse noise and lacks of computational cost. In [2] a new edge-based static external force is proposed called Vector Field Convolution (VFC). The basic idea behind VFC-snakes is to convolve a specific vector field kernel with the edge map derived from the image, whereas the vector field strongly forces the snake towards the edges.
The vector field kernel $\underline{k}(\underline{v})$ is defined by the equation

$$\underline{k}(\underline{v}) = m_n(\underline{v}) \frac{-\underline{v}}{|\underline{v}|} \qquad (18)$$

where $m_n(\underline{v})$ is the vector magnitude function of the vector at (x,y) and $\frac{-\underline{v}}{|\underline{v}|}$ is the unit vector pointing to the kernel origin at (0,0) as the centre of the kernel. The magnitude function incorporates the distance from the origin and decreases the influence of edges if the edge elements are further away. This property is fulfilled by the given equations

$$m_1(\underline{v}) = (|\underline{v}| + \varepsilon)^{-\gamma} \qquad (19)$$

$$m_2(\underline{v}) = e^{(\frac{-|\underline{v}|^2}{\zeta^2})} \qquad (20)$$

where γ and ζ are used to control the decrease of ROI influence and ε denotes a small positive constant to prevent division by zero at the origin [2].
The VFC external force is computed by convolving the edge map with the vector field kernel given by the equation

$$\underline{F}_{vfc} = f(\underline{v}) \otimes \underline{k}(\underline{v}) \qquad (21)$$

where $f(\underline{v})$ is the edge map. To support also the detection of weak edges, a mixed VFC-field was proposed by [2] given by the equation

$$\underline{F}_{ext}(\underline{v}) = \underline{F}_{mix}(\underline{v}) = \begin{cases} \nabla \underline{f}, & |\nabla \underline{f}| > \phi \\ \underline{F}_{vfc}, & |\nabla \underline{f}| \le \phi \end{cases} \qquad (22)$$

where ϕ is a smoothing parameter like μ of GVF and $\nabla \underline{f}$ denotes the gradient of an edge map. As a result the mixed VFC-field can be considered as the standard external force mixed with the VFC-field in homogenous regions depending on ϕ.

3. Extending the capture abilities and speed of VFC-snakes

3.1. The new Alternating Vector Field Convolution Snakes (AVFC)

With regard to [2], the VFC-snakes have better capture capabilities, are more robust to noise, are flexible when tailoring the force field and concurrently reduce computational cost for approximating the desired ROI. For the examples published in literature this assumption should be supported. But also in some specific cases for instance approximating a simple circle starting from inside the circle balloon snakes converge faster with the ROI. This follows from increased computational cost if large images are convolved with the filter kernel (21) independent of the method used for convolving the image.
According to [10], where GVF-snakes are extended with specific balloon forces to increase convergence speed, an extended approach for VFC-snakes is contributed in the following.
The convergence speed of snakes at the beginning of an iterative computation is dependent on the magnitudes of the vector fields used and is usually rather low. This could be improved by applying additional balloon forces on VFC-snakes, which concurrently restricts the snake approach to a closed loop.
To avoid distortions of the VFC curve evolution, it is important to dynamically adjust the balloon extension with regard to the current deform iteration which gives the following equation for the snaxels based on (10) to

$$\underline{v}_t = \left(\underline{A} + \gamma \underline{I} \right)^{-1} \left(\gamma \underline{v}_{t-1} + \frac{k_1}{\sqrt{i}} \underline{n}(\underline{v}) - k \frac{\underline{F}_{mix}(\underline{v})}{|\underline{F}_{mix}(\underline{v})|} \right) \qquad (23)$$

Depending on the choice of k_1 the snake can be now inflated or deflated with respect to both the current deform iteration i and snake iteration t and, consequently, may simultaneously converge faster. The choice of k_1 in equation (23) directly influences the behavior of the snake and always requires an user interaction. This is an undesirable property if the edge detection method is to work automatically. The decision on whether the snake needs to be inflated starting from the inner image centre towards the edges or deflated vice versa is also complex. To minimize the decision process, the use of Multi Resolution Analysis (MRA) and an alternating balloon force is suggested.

To prevent balloon forces producing unpredictable results [11], the snake should be inflated in the first and coarsest resolution starting from the centre to extract the inner shape. In the next resolution level the snake should start from the previous snake curve and then be deflated towards the image centre and so on.

This alternating approach concurrently respects possible shifts of real edges caused by resampling methods within the MRA and increases the speed of the whole edge detection process [12]. Defining res_j with $j \in [0,N]$ as the square root of the image size respecting MRA level j where 0 denotes the highest level and N the lowest level considered as the original image the equation (23) changes to

$$\underline{\mathbf{v}}_{t,j} = \left(\underline{\mathbf{A}}_j + \gamma \underline{\mathbf{I}}_j\right)^{-1} \left(\gamma \underline{\mathbf{v}}_{t-1,j} + \frac{-1^j \, k_1 \, res_0}{res_j \, \sqrt{i_j}} \, \underline{\mathbf{n}}_j(\mathbf{v}) - \right.$$
$$\left. k \, \mathbf{F}mixj(\mathbf{v}) \mathbf{F}mixj(\mathbf{v}) \right) \qquad (24)$$

Thus, the edge detection speed in the original image resolution increases by the initial contour estimation gathered from higher MRA-levels. This property is essential for large images.

To avoid computation time intensive inversion of matrix $\left(\underline{\mathbf{A}} + \gamma \underline{\mathbf{I}}\right)$ which size depends on the number of snaxels, [12] proposed a method to subdivide this matrix into 4 pieces. The solution for all pieces can be computed by using an optimized Cholesky decomposition. The speed limitation to this approach is the use of for-loops which slow the edge detection process for large images and can concurrently override available computer memory unless for-loops are usually consecutively saved in local memory. Based on [6] the inversion of matrix $\left(\underline{\mathbf{A}} + \gamma \underline{\mathbf{I}}\right)$ can be completely avoided. $\underline{\mathbf{A}}\underline{\mathbf{v}}_t$ is approximated by $\underline{\mathbf{A}}\underline{\mathbf{v}}_{t-1}$ which then gives the equation for the extended and mixed VFC-snake and is addressed as Alternating Vector Field Convolution Snakes (AVFC). The related equation is given to:

$$\underline{\mathbf{v}}_{t,j} = -\frac{1}{\gamma} \left((\underline{\mathbf{A}}_j + \gamma \underline{\mathbf{I}}_j) \underline{\mathbf{v}}_{t-1,j} + \frac{-1^j \, k_1 \, res_0}{res_j \, \sqrt{i_j}} \, \underline{\mathbf{n}}_j(\mathbf{v}) - \right.$$
$$\left. k \, \frac{\underline{\mathbf{F}}_{mix\,j}(\mathbf{v})}{\left|\underline{\mathbf{F}}_{mix\,j}(\mathbf{v})\right|} \right) \qquad (25)$$

The AVFC can be imagined as a slowly growing and pulsating figure in 2D or bubble in 3D which delineates the region of interests. Curve evolving properties such as curvature, stiffness etc. can also be considered and will lead to parameterizations of e.g. α with respect to time t and res_j.

3.2. The new Yukawa Alternating Vector Field Convolution Snakes (YAVFC)

According to [2] the capture capability of a snake algorithm depends on the magnitude functions given by equations (19) and (20). As the basic property of the magnitude functions is to strongly decrease the magnitude with respect to the distance to the centre, any function which fulfills this requirement can be used.

This is also fulfilled by the distance function based on the famous Yukawa Potential [13], which is related to the forces in equal charged particles. Those forces well-known as Strong Forces belong to the elementary forces considered in Nuclear Physics. As a special case of the Yukawa Potential the magnitude function

$$m_{Yuk}\left(\underline{\mathbf{v}}\right) = \frac{e^{-|p\,\underline{\mathbf{v}}|}}{|\underline{\mathbf{v}}| + \varepsilon_1} \qquad (26)$$

fulfils the requirements for a force field to move free particles towards the edges where p is a small geometry factor and ε_1 is a small, positive constant to prevent division by zero at the origin. Substituting the vector field kernel based on the equations (19 or 20) with equation (26) gives a new external force called Strong Force or Yukawa Force.

The related Active Contour is addressed as Yukawa Alternating Vector Field Convolution Snake (YAVFC), belongs to the AVFC snakes and is based on the equation

$$\underline{\mathbf{v}}_{t,j} = -\frac{1}{\gamma} \left((\underline{\mathbf{A}}_j + \gamma \underline{\mathbf{I}}_j) \underline{\mathbf{v}}_{t-1,j} + \frac{-1^j \, k_1 \, res_0}{res_j \, \sqrt{i_j}} \, \underline{\mathbf{n}}_j(\mathbf{v}) - \right.$$
$$\left. k \, \frac{f_j(\underline{\mathbf{v}}) \otimes \frac{-\underline{\mathbf{v}}}{|\underline{\mathbf{v}}|^2 e^{|p\,\underline{\mathbf{v}}|} + \varepsilon_2}}{\left| f_j(\underline{\mathbf{v}}) \otimes \frac{-\underline{\mathbf{v}}}{|\underline{\mathbf{v}}|^2 e^{|p\,\underline{\mathbf{v}}|} + \varepsilon_2} \right|} \right) \qquad (27)$$

where ε_2 is a small, positive constant to prevent division by zero at the origin. Concerning the property of the VFC external force to be a vector field with a negative divergence or a sink which is the underlying idea of VFC-snakes to force free particles towards the edges, this principle is similar for the YAVFC-snakes. With regard to the associativity and differentiation rules for partial derivatives of the convolution the divergence of YAVFC external force $\underline{\mathbf{F}}_{YAVFC}\left(\underline{\mathbf{v}}\right)$ is given by the following equation

$$\nabla \underline{\mathbf{F}}_{YAVFC\,j}\left(\underline{\mathbf{v}}\right) = \nabla \left(f_j\left(\underline{\mathbf{v}}\right) \otimes \frac{-\underline{\mathbf{v}}_j}{|\underline{\mathbf{v}}_j|^2 e^{|p\,\underline{\mathbf{v}}_j|} + \varepsilon_2} \right) =$$
$$\frac{-f_j(\underline{\mathbf{v}})}{|\underline{\mathbf{v}}_j|^2 e^{|p\,\underline{\mathbf{v}}_j|} + \varepsilon_2} \otimes \nabla\left(\underline{\mathbf{v}}_j\right) = \frac{-f_j\left(\underline{\mathbf{v}}_j\right)}{|\underline{\mathbf{v}}_j|^2 e^{|p\,\underline{\mathbf{v}}_j|} + \varepsilon_2} \, tr(\underline{\mathbf{I}}) \qquad (28)$$

where ∇ denotes the divergence, $\otimes$ the convolution, $f_j\left(\underline{\mathbf{v}}\right)$ the MRA edge map and $tr\left(\underline{\mathbf{I}}\right)$ the trace of a dimension

related identity matrix. Assuming non negative values for the edge map $f_j(\underline{\mathbf{v}})$, this new external force fulfils the basic requirements of being a sink for edges.

With regard to (28) it is also possible to increase the influence of weak edges by multiplying the edge map with non-linear platykurtic, intensity related functions or to increase the influence of strong edges by multiplying the edge map with non-linear leptokurtic, intensity related functions. The direct use of the Yukawa Force (26) by multiplying with the gradient of an edge map can be considered as another approach to enhance the edge detection algorithms based on usual snake methods.

3.3. Noise removal

In specific cases remaining impulse noise still distorts the edge detection and should be removed. Usually, a linear filter, for instance a Gaussian Filter or a non-linear filter, for instance the Median Filter is applied on the image to reduce remaining noise. The linear filter class reduces noise, but the blurring and smoothing of the edges can hinder the precise localization of edges if the edges are rather thin. The non-linear filter class reduces noise better, but shifts the edges somehow depending on the edge itself and the filter width. Those properties are quite complex to model and even not useful for an effective edge detection method based on snakes.

Another way to minimize noise in images is proposed in [14,15]. Based on [14,15] adaptive, data-driven Coiflet hard-thresholds are applied on the image with respect to the BayesShrink rules. By [16] this method shall not distort edges, shall not produce grainier filter results and seems to be suitable for image denoising.

3.4. Edge map

In [2] an edge map of a gray-level image is computed e.g. as the negative, squared norm of a Gaussian filtered image (3) or as inverse of the intensity function (2) for 1-Bit-images. The edge map is widely used as input for the most snake approaches. With regard to [17,18,19] a good edge detector must detect and localize edges and shall only allow single edge responses. These properties are given by the Canny edge detection method [17,19] and therefore, this method was applied to compute an edge map of an image.

In some specific cases the canny filter tended to overestimate heterogeneous image regions as edges across a real edge depending on the choice of the hysteresis thresholds and the noise parameter σ. Reasoned by this property morphological dilations with small, rectangular discs as structuring elements were applied. The dilation closes small gaps along and across the edges with respect to the disc size [20].

3.5. Speed issues

[12] propose specific approaches to speed up snakes by, for example parallelization of computing processes, MRA and specific decompositions to invert the pentadiagonal matrix $\underline{\mathbf{A}} + \gamma\underline{\mathbf{I}}$. [2] suggests to use the Fast Fourier Transform (FFT) for the convolution of the VFC filter kernel with the edge map instead of linear convolution. Extending this approach it is possible to generally substitute discrete linear convolution with FFT in all snake methods [21]. This can also increase the computing speed of gradients, edge maps etc. Reducing the calculation time is essential for e.g. motion tracking, so it is also important to break the snake deformations and iterations if a certain threshold has been reached. Such a threshold is usually related to the difference or rate of change between the i-th and the (i-1)-th deformation or iteration.

Setting $\Delta\underline{\mathbf{v}}_t = \underline{\mathbf{v}}_t - \underline{\mathbf{v}}_{t-1}$ as the difference between the deformation results in the following equation for an reliable threshold to

$$\left| \Delta\underline{\mathbf{v}}_t^{\mathrm{T}}[\mathrm{diag}(\underline{\mathbf{I}})^{\mathrm{T}}, \mathrm{diag}(\underline{\mathbf{I}})^{\mathrm{T}}] \right| \begin{cases} \geq \text{threshold, continue} \\ < threshold,\ break \end{cases} \quad (29)$$

Equation (29) can also be used when iterating each snake, not only when deforming it. The prior equation is similar to the convergence criterion proposed by [1] which is related to the Courant-Friedrichs-Lewy step-size restriction depending on the threshold chosen e.g. in (29). Another method to define the threshold for the iterations is the use of the change rate of the surrounded polygonal area by the snake. If the number of snaxels do not change significantly within the iterations, it is also not necessary to compute matrix $\underline{\mathbf{A}} + \gamma\underline{\mathbf{I}}$ (25, 27) again, because it has not changed.

4. Results

4.1. Quality

To test and to evaluate different snake approaches, 8-Bit rectangular, real and artificial sample images with different image sizes were chosen. Sample 1 was a 128 x 128 pixel sized part of a remotely sensed image (Fig. 1 a). Sample 2 and 5 were 256 x 256 pixel sized form primitives (Fig. 1 b and e). Sample 3 was a 64 x 64 pixel sized part of a remotely sensed image (Fig. 1 c). Sample 4 was a 128 x 128 pixel sized magnetic resonance image of the left ventricle of a human heart (Fig. 1 d).
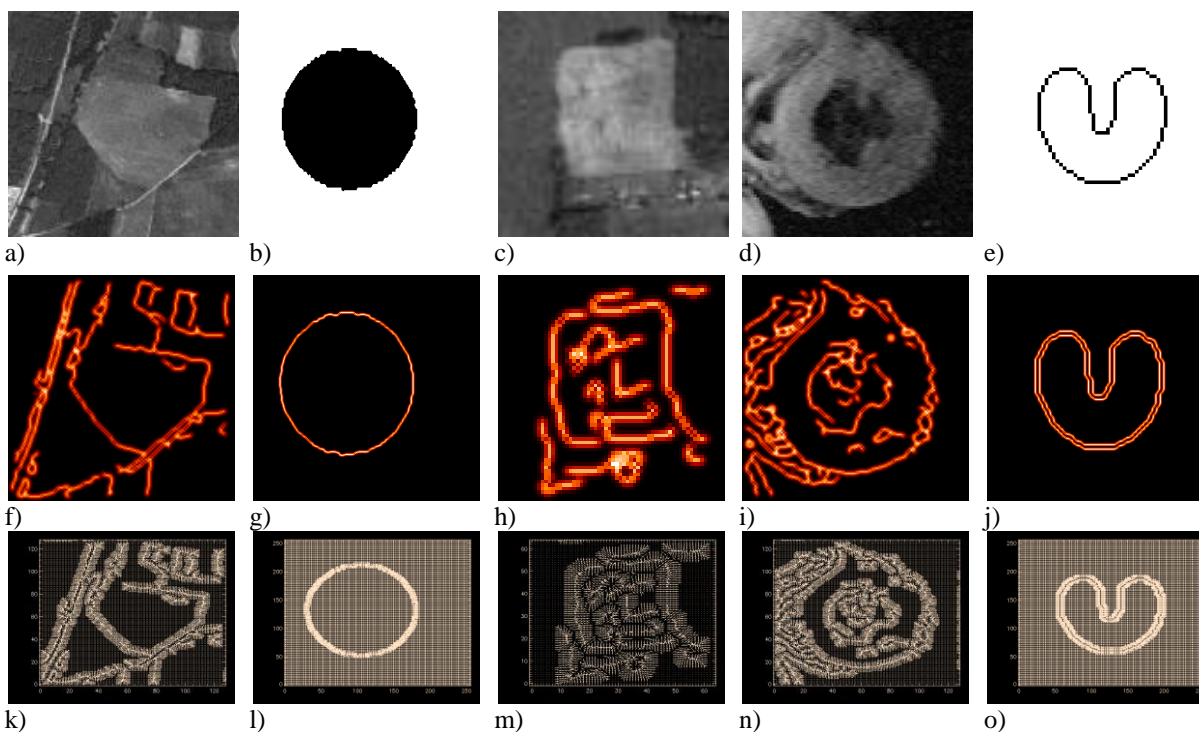
Fig. 1   Sample images (a-e), edge maps (f-j), gradient fields of classic and Balloon snakes (k-o)

At first all images were Coiflet based BayesShrink hard-thresholded to reduce noise. Subsequent to the image denoising the edge maps were computed by applying the canny algorithm (Fig. 1 f-j). To remove small junctions and gaps in near to strong edges, morphological dilations with 2x2 rectangular discs as structuring elements were applied on the edge maps.

In a next step remaining impulse noise was minimized by convolving the edge maps with a linear 2D-Gaussian filter kernel. To enable mixed external force fields (22), the gradients of the edge maps were additionally computed (Fig. 1 k-o).

Then, small, centered circles were used as initial contours for different snake algorithms in different images (Fig. 2 a-e) which is similar to the initializations published in literature. Starting from initial contours each snake was deformed 5 times within 500 iterations with the same parameter sets to ensure comparability.

The external forces of the different snake approaches were computed in relation to their corresponding snake equations given before.

According to [2] classic and Balloon snakes need sometimes initial contours relatively near the target contour to converge. This property was also observed and hence, they were not compared, as these approaches had to have other initial conditions such as GVF-snakes or did not converge based on the given initial contour in time. It follows from this that the GVF, the VFC, the AVFC and the YAVFC snakes were compared (Fig. 2).
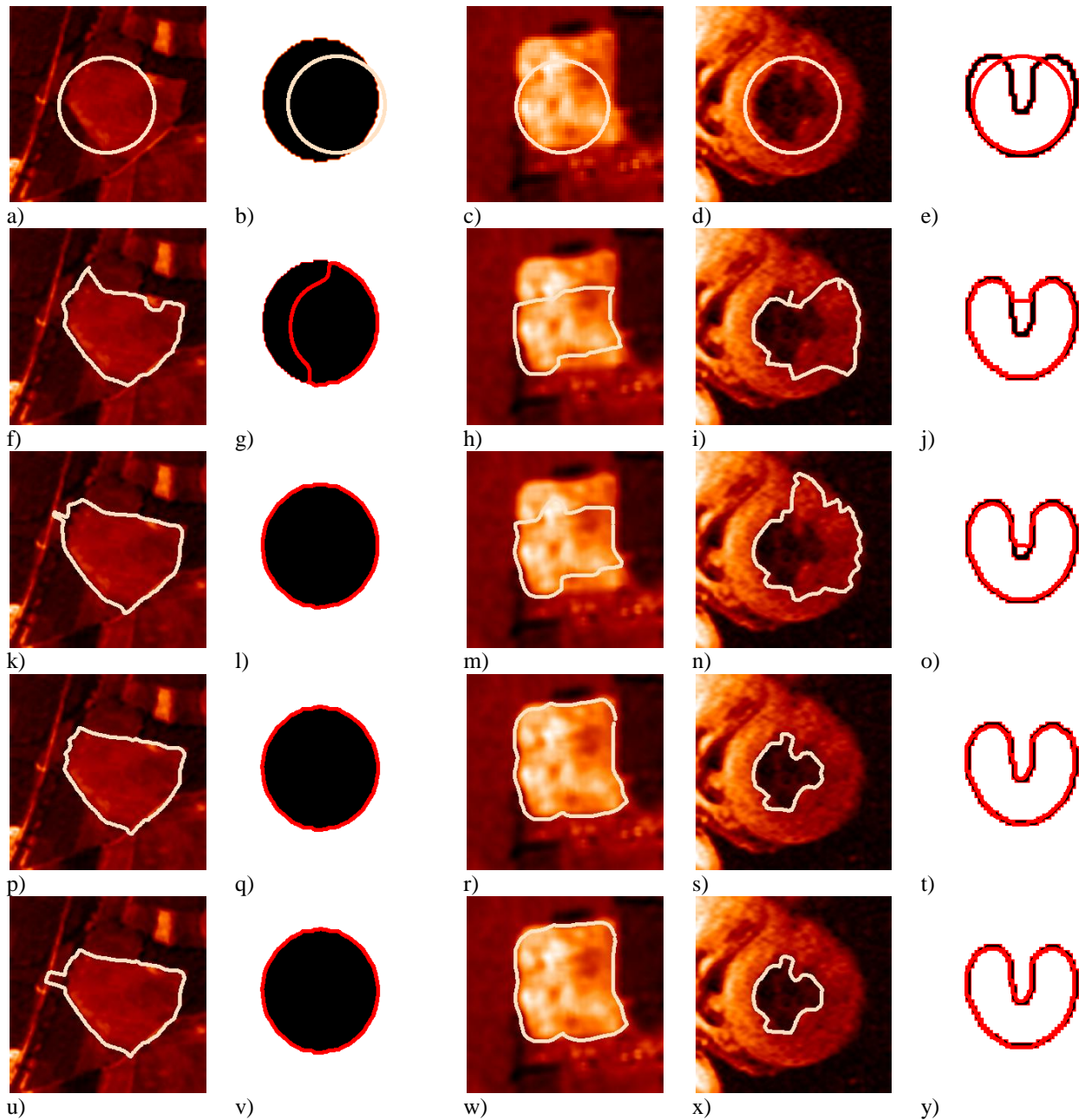
Fig. 2   Initial contours on false colour sample images (a-e), results of GVF-snake (f-j), results of VFC-snake(k-o), results of AVFC-snake (p-t), results of YAVFC-snake(u-y)

With regard to the results given in (Fig. 2 f-j), the GVF-snakes were not able to converge completely with simple figures (Fig.2 g and j) conditioned by the amount of iterations used. This was also shown by [2] and can be supported. It's also noticeable, that GVF-snakes tend to overestimate weak edges (Fig. 2 f,h,i) and hence, despite their smoothing abilities the GVF snakes in the original formulation may only have specific use.

VFC-snakes (Fig. 2 k-o) contribute better results in comparison to GVF-snakes with regard to given form primitives (Fig. 2 l and o) and enhance edge detection in specific samples (Fig. 2 k). Conditioned by the amount of iterations, VFC-snakes expand and move faster into concavities (Fig. 2 o) or in general towards boundaries (Fig. 2 m and n). They are flexible in usage and can be better adjusted than GVF-snakes. Therefore, they are

useful in a broader environment, but have to be finely adjusted.

In relation to the results (Fig. 2 p-y) AVFC- and YAVFC snakes provided the best results and were able to capture all ROI independent of the given samples. This is caused for both the AVFC- and the YAVFC-snakes by the oscillations of parameters and different forces, but also conditioned by the Multi Resolution Analysis approach.

Comparing the AVFC- with the YAVFC-snake, it can be assumed that the YAVFC-snake is more dependent on parameterizations, but produces smoother results and tends to move the snake more towards strong edges compared to the AVFC-snake. In contradiction to the YAVFC snake the AVFC-snake provides better results with regard to edges near the ROI. Concerning those results and the overall capture quality, the use of the new AVFC-snake and its derivative YAVFC-snake based on a completely new Force called Strong or Yukawa Force is suggestible for a broader range of applications.

4.2. Speed

To enable a broad range of applications, the convergence speed must also be considered. This is also a quality criterion.

In relation to given samples AVFC and YAVFC snakes converge much faster on average (10 trials per sample) than GVF- and VFC-snakes (Table 1). This property is strongly related to the size of the image i.e. the scalar field. In some specific cases YAVFC-snake outperforms the AVFC-snake. For small image sizes VFC-snakes are also very fast, which is conditioned by the amount of computations in relation to YAVFC and AVFC snakes. This advantage is relevant until images are very small. In summary, AVFC and YAVFC snakes are more than 400 times faster than GVF and VFC-snakes and should be applied for most edge detection cases. This benefit will be enforced with increasing image size.

Table 1: Runtime for different Active Contour approaches

	Sample 1 on average in seconds	Sample 2 on average in seconds	Sample 3 on average in seconds	Sample 4 on average in seconds	Sample 5 in on average seconds	Total average time consumption in seconds
Rectangular sample size	128	256	64	128	256	
GVF	76.5	350.2	18.1	92.9	399.9	937.5
VFC	98.6	592.5	1.1	67.4	523.9	1283.5
AVFC	0.4	0.61	0.5	0.3	**0.3**	2.3
YAVFC	**0.1**	**0.59**	**0.4**	**0.3**	0.6	**2**

*all results were achieved on a Intel Core2Duo L7500 X61s ThinkPad with 4 GB RAM within MS Windows Vista X64 Business and Interactive Data Language (IDL) 6.4 x64

With regard to the suggestion of [2] to use complex FFT for the convolution instead of discrete convolution, it is also necessary to test the proposed speed and to compare it.

To compare linear convolution with FFT convolution, averaged trials (100 trials for each case) for different matrix and kernel sizes were computed with regard to the convolution method (Table 2).

Table 2: Runtime for different convolution approaches

Matrix – size in both directions	Kernel – size in both directions	Linear convolution on average in seconds	FFT convolution on average in seconds	Ratio - FFT / Linear on average
32	32	0.0166	**0.0003**	60
64	64	0.2520	**0.0008**	300
128	128	3.9351	**0.0033**	1300
128	3	0.0395	**Not measureable**	Not measureable
256	3	0.6247	**0.0002**	3000
512	3	10.1722	**0.0003**	32485

*all results were achieved on a Intel Core2Duo L7500 X61s ThinkPad with 4 GB RAM within MS Windows Vista X64 Business and Interactive Data Language (IDL) 6.4 x64

According to [21] convolution speed of FFT is significantly higher than linear convolution at each tested matrix and kernel size (Table 2). From this it follows that

the convolution of matrices e.g. for the computation of external forces, gradients etc. will be faster by using FFT.

4.3. Implementation

All the methods described before are fully implemented in Interactive Data Language (IDL) which offers a broad range for applications. All IDL implementations of above algorithms work autonomously and automatically. They are cross-platform useable by the existing IDL runtime environment.

5. Conclusions

In previous chapters new methods based on new approaches and based on a new force were introduced and compared to usual methods. With regard to the results gathered the AVFC and the YAVFC snake outperform considered methods in terms of speed and quality. These properties are evident and the results can be repeated. According to this a recommendation of proposed new approaches will be concluded. In a next step it may be necessary to transform the proposed closed snake approaches into open-ended to enable a broader set of applications.

References

[1] Xu, C. and Prince, J.L. ,1998. Snakes, shapes, and gradient vector flow. IEEE Transactions on Image Processing 7, 359-369.

[2] Li, B. and Acton, S.T. ,2007. Active contour external force using vector field convolution for image segmentation. IEEE Transactions on Image Processing 16, 2096-2106.

[3] Kass, M., Witkin, A. and Terzopoulos, D. ,1988. Snakes: Active contour models. International Journal of Computer Vision V1, 321-331.

[4] Cohen, L. and Cohen, I. ,1993. Finite-element methods for active contour models and balloons for 2-D and 3-D images. IEEE Transactions on pattern analysis and machine intelligence 15, 1131-1147.

[5] Amini, A., Weymouth, T. and Jain, R. ,1990. Using Dynamic Programming for Solving Variational Problems in Vision. IEEE Transactions on Pattern Analysis and Machine Intelligence 12, 855-867.

[6] Cohen, L. ,1991. On active contour models and balloons. Computer Vision, Graphics, and Image Processing. Image Understanding 53, 211-218.

[7] Xu, C. and Prince, J.L. ,1997. Gradient vector flow: a new external force for snakes. , CVPR '97: Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97), 66-71.

[8] Horn, B. and Schunck, B. ,1981. Determining optical flow. Artificial Intelligence 17, 185-203.

[9] Xu, C. and Prince, J.L. ,1998. Generalized gradient vector flow external forces for active contours. Signal Processing 71, 131-139.

[10] Luo, S., Li, R. and Ourselin, S. ,2003. A New Deformable Model Using Dynamic Gradient Vector Flow and Adaptive Balloon Forces. APRS Workshop on Digital Image Computing, Brisbane, Australia.

[11] Ballangan, C. ,2005. Comparison of three different image forces for active contours on abdominal image boundary detection. Jurnal Informatika 6, 71-75.

[12] Kienel, E., Vanco, M. and Brunnett, G. ,2006. Speeding up snakes. VISAPP 2006: Proceedings of the First International Conference on Computer Vision Theory and Applications, 1, 323-330.

[13] Yukawa, H. ,1935. On the Interaction of Elementary Particles. Proc. Phys.-Math. Soc. 17, 48-57.

[14] Chang, S.G., Yu, B. and Vetterli, M. ,2000. Adaptive wavelet thresholding for image denoising and compression. IEEE Transactions on Image Processing 9, 1532-1546.

[15] Mohideen, S.K., Perumal, S.A. and Sathik, M.M. ,2008. Image De-noising using Discrete Wavelet transform. IJCSNS International Journal of Computer Science and Network Security 8, 213-216.

[16] Fodor, I.K. and Kamath, R. ,2003. Denoising through wavelet shrinkage: an empirical study . SPIE Journal on Electronic Imaging 12, 151-160.

[17] Canny, J. ,1986. A computational approach to edge detection. IEEE Transactions on pattern analysis and machine intelligence 8, 679-698.

[18] Ding, L. and Goshtasby, A. ,2001. On the Canny edge detector. Pattern Recognition 34, 721-725.

[19] Sharifi, M., Fathy, M. and Mahmoudi, M.T. ,2002. A Classified and Comparative Study of Edge Detection Algorithms. ITCC '02: Proceedings of the International Conference on Information Technology: Coding and Computing, 8-10 April 2002, 117 - 120.

[20] Haralick, R.M., Sternberg, S.R. and Zhuang, X. ,1987. Image analysis using mathematical morphology. IEEE Transactions on Pattern Analysis and Machine Intelligence 9, 532-550.

[21] Smith, J.O. ,2007. Mathematics of the Discrete Fourier Transform (DFT) with Audio Applications, Second Edition, W3K Publishing, http://books.w3k.org/, ISBN 978-0-9745607-4-8.