# A MESSAGE PASSING ALGORITHM FOR ACTIVE CONTOURS

*Ahmed Kirmani* *, *Naveen Goela* †, *Niladri Chatterjee* ‡, *Ben Vigoda* §

## ABSTRACT

Many important early vision techniques, such as Active contours (ACs), can be formulated as energy minimization. However, finding global optimum (minimum energy) configurations is often computationally intractable. Approximate solutions obtained using iterative numerical methods may be ill-conditioned, and exhibit poor convergence and inaccuracy due to noise and discretization errors. We formulate AC as a statistical estimation problem and solve it using (Gaussian) Message passing on Factor graphs of linear models. The resulting algorithm exhibits faster convergence and the solutions possess higher numerical stability, robustness and accuracy.

***Index Terms*—** Message passing, Machine Vision, Linear systems, Optimization, Statistical Estimation.

## 1. INTRODUCTION

Active contours (ACs) are popular methods for solving a variety of vision problems. ACs are energy-minimizing splines controlled by internal constraint forces and external (image) forces that pull it towards features, such as lines and edges. The energy functions used in ACs are typically non-convex, and the space of possible configurations is very large. They have many local minima which may be arbitrarily far from the optimum. Hence finding optimum solutions to the corresponding optimization problem is computationally intractable. Another common issue is that local minimization techniques are naturally sensitive to the initial estimate.

Various techniques exist for energy minimization in ACs. Dynamic programming on curves, Neural networks and (Euler-Lagrange) numerical methods [1] are widely used to minimize the discrete energy functional in ACs. Fig. 3(a), 3(b), 3(c) show the use of Dynamic programming for AC. Euler-Lagrange method achieves this minimization iteratively by solving an inverse problem and suffers from numerical instability because numerical differentiation is ill-posed when the solution does not depend continuously on the data.

Many known algorithms for practical applications in coding, artificial intelligence, and signal processing may be viewed as instances of the Sum(mary)-Propogation Algorithm (SPA) that operates by Message passing on Factor graphs (FGs) [2], [3], [4]. Gaussian message passing in FGs of linear models was considered in [5]. Many unconstrained optimization techniques can be elegantly combined with SPAs to yield powerful tools for statistical estimation.

We formulate discrete energy minimization in ACs as an instance of Gaussian Message passing over FGs of linear models and combine the robustness and computational efficiency of SPA to achieve faster, accurate and robust convergence in presence of noise or an ill-conditioned problem. This approach offers significant gains over the (Euler-Lagrange) numerical methods for ACs (snakes) which are highly sensitive to noise, false minima and contour initialization.

The rest of the paper is organized as follows: Section 2 is a primer for the Euler-Lagrange method for ACs and FGs. Section 3 serves as the motivation to derive an iterative Message passing algorithm for ACs and discusses the novel formulation of AC as an instance of SPA involving Gaussian Message passing in a FG of linear models. Section 4 discusses the simulations and the results of implementation of different algorithms. Section 5 offers some conclusions and discusses potential application of the technique developed in this paper to other numerical algorithms.

## 2. BACKGROUND

Active contours can be formulated as a function $\mathbf{v} : [0, 1] \rightarrow \mathbb{R}^2$ with some boundary conditions. The contour is placed on an image $\mathbf{I} : \mathbb{R}^2 \rightarrow \mathbb{R}$, and it moves towards an *optimal* position and shape by minimizing its own energy. Fitting active contours to shapes in images may be stated as finding $\hat{\mathbf{v}} = \text{argmin}_{\mathbf{v}} E_{contour}(\mathbf{v}, \mathbf{I})$. Parametrically the AC can be represented as $\mathbf{v}(s) = (\mathbf{x}(s), \mathbf{y}(s))$, and its energy functional is $E_{contour}(\mathbf{v}(s), \mathbf{I}) = E_{int}(\mathbf{v}(s)) + E_{image}(\mathbf{v}(s), \mathbf{I})$. The internal energy $E_{int}$ of the contour depends on the shape of the contour and the parameter functions $\alpha(s)$ (elasticity) and $\beta(s)$ (rigidity). The image energy $E_{image}$ is a function of the contour position on an attractor image $p(\mathbf{v}(s), \mathbf{I})$. Examples of image functionals are the image intensity $E_{line} = \mathbf{I}(x, y)$ and $E_{edge} = -|\nabla \mathbf{I}(x, y)|^2$. These energies result in internal and external (image) forces at all points of the contour. When all forces are balanced, the total energy is at a minimum.

Using variational calculus and by applying Euler-Lagrange differential [1] shows that minimizing $E_{contour}$ gives rise to

*Department of Mathematics, Indian Institute of Technology Delhi. Author thanks Lyric Semiconductor Inc. MA, USA for supporting this research.
†EECS, University of California Berkeley CA 94720, USA.
‡Department of Mathematics, Indian Institute of Technology Delhi.
§Lyric Semicondutor Inc. Cambridge, USA and MIT Media Laboratory

the following force balance equation,

$$\underbrace{\alpha\frac{\partial^2\mathbf{v}(s)}{ds^2} - \beta\frac{\partial^{(4)}\mathbf{v}(s)}{ds^{(4)}}}_{\text{Internal}-\text{Contour}-\text{force}} = \underbrace{-\gamma\nabla E_{ext}}_{\text{External}-\text{Image}-\text{force}}$$

where $\gamma$ is a constant (step size) introduced to control the external force. The discrete formulation of the contour is a piecewise linear curve obtained by joining a set of control points $\mathbf{v} = v_0, v_1, \ldots, v_{n-1}, v_n)$ where $v_i = (x_i, y_i)$ and $v_n = v_0$ (closed contour). The discrete energy terms are:

$$\begin{aligned} E_{contour}(\mathbf{v},\mathbf{I}) &= \sum_{i=1}^{n} E_{int}(v_i) + \sum_{i=1}^{n} E_{img}(v_i,\mathbf{I}) \\ E_{int}(v_i) &= \underbrace{\frac{\alpha_i|v_i - v_{i-1}|^2}{2h^2}}_{Elasticity} + \underbrace{\frac{\beta_i|v_{i-1} - 2v_i + v_{i+1}|^2}{2h^4}}_{Stiffness} \\ E_{img}(v_i,\mathbf{I}) &= -\left(|G_x(x_i,y_i)|^2 + |G_y(x_i,y_i)|^2\right) \end{aligned}$$

where $G_x = \frac{\partial G_\sigma}{\partial x} \otimes \mathbf{I}$, $G_y = \frac{\partial G_\sigma}{\partial y} \otimes \mathbf{I}$, $G_\sigma$ is the convolution Gaussian of standard deviation $\sigma$ and $\otimes$ is the convolution operator. Let $f_x(i) = \partial E_{img}/\partial x_i$ and $f_y(i) = \partial E_{img}/\partial y_i$ where the derivatives are approximated by finite differences when they cannot be computed analytically. Each control point is allowed to move freely under the influence of the forces. The discrete force balance (constraint) equation $C_{v_i}$ at each point $v_i$ is,

$$\begin{aligned} &\alpha_i(v_i - v_{i-1}) - \alpha_{i+1}(v_{i+1} - v_i) \\ &\quad + \beta_{i-1}(v_{i-2} - 2v_{i-1} + v_i) \\ &\quad - 2\beta_i(v_{i-1} - 2v_i + v_{i+1}) \\ &\quad - \beta_{i+1}(v_i - 2v_{i+1} + v_{i+2}) + (f_x(i), f_y(i)) = 0 \end{aligned}$$

which can be written as $\mathbf{A}^i\nu_i + \mathbf{f}(v_i) = 0$, where $\mathbf{A}^i$ is a coefficient row vector and $\nu_i = [v_{i-2}, v_{i-1}, v_i, v_{i+1}, v_{i+2}]^T$. Then the discrete force balance equation for the contour is $\mathbf{A}\mathbf{v}+\mathbf{f}(\mathbf{v}) = 0$, where $\mathbf{A}$ is a pentadiagnol banded coefficient matrix. The solution to this equation is given by an implicit Euler step involving time derivatives,

$$\mathbf{A}\mathbf{v}^t + \mathbf{f}(\mathbf{v}^t) = -\gamma(\mathbf{v}^t - \mathbf{v}^{t-1}) \tag{1}$$

At equilibrium, the time derivative vanishes and we end up with a solution of the force balance equation. Equation 1 can be solved by matrix inversion $\mathbf{v}^t = (\mathbf{A} + \gamma\mathbf{I})^{-1}(\mathbf{v}^{t-1} - \mathbf{f}_{\mathbf{v}^{t-1}}(\mathbf{x}^{t-1}, \mathbf{y}^{t-1}))$. The matrix $(\mathbf{A} + \gamma\mathbf{I})$ is a pentadiagonal banded sparse matrix, so its inverse can be calculated by LU decompositions in $O(n)$ time and hence provides a rapid solution to (1).

Factor graphs represent the factorization of a global function into a product of local factors, $f(\mathbf{X}) = \prod_{j=1}^{m} f_j(S_j(\mathbf{X}))$ where, $S_j(X) \subset \mathbf{X} = \{x_1, \ldots, x_n\}$. While graphically representing factor graphs, we use circles for function nodes $f_j$ and squares for variable nodes $x_i$. There is an edge that connects each function node to each of the variable nodes in its arguments. The global function $f$ is computed by applying SPA which states that, the message sent from a node $v$ on an edge $e$ is the product of the local function at $v$ (or the unit function if $v$ is a variable node) with all messages received at $v$ on edges other than $e$, summarized for the variable associated with $e$. Let $\mu_{x\to f}(x)$ be the message passed from variable node $x$ to function node $f$, and similarly $\mu_{f\to x}(x)$. We have,

$$\begin{aligned} \mu_{x\to f}(x) &= \prod_{h\in\mathcal{N}(x)\setminus\{f\}} \mu_{h\to x}(x) \\ \mu_{f\to x}(x) &= \sum_{X\setminus x}\left(f(X_{\mathcal{N}(f)}) \prod_{y\in\mathcal{N}(f)\setminus\{x\}} \mu_{y\to f}(y)\right) \end{aligned}$$

where $\mathcal{N}(.)$ is the neighborhood function. The reader should refer to prefatorial papers on snakes [1] and Factor graphs [2] for complete descriptions and derivations.

## 3. MESSAGE PASSING FORMULATION

The following iterative Euler approximation is frequently used $\mathbf{v}^t \simeq \frac{1}{\gamma}\{\mathbf{v}^{t-1} - (\mathbf{A}\mathbf{v}^{t-1} + \mathbf{f}(\mathbf{v}^{t-1}))\}$, where $\mathbf{v}^0$ is the set of initially suggested control points. This can be also written as $\gamma\mathbf{v}^t = \mathbf{v}^{t-1} + \Delta^t$ where at any time (iteration) $t$, let $\Delta^t = \Delta(\mathbf{v}^{t-1}) = \mathbf{A}\mathbf{v}^{t-1} + \mathbf{f}(\mathbf{v}^{t-1})$ be an update function. $\Delta^t = (\Delta_1^t, \ldots, \Delta_n^t)$ can be viewed as series of *local* updates at the control points i.e. $v_i^t = v_i^{t-1} + \Delta_i^t$ , where, $\Delta_i^t = \sum_{j=-2}^{2} \mathbf{A}_{i+j}^i v_{i+j}^{t-1} + \mathbf{f}(v_i^{t-1})$.

Note that $C_v = C_v(\mathcal{N}(v), f(\mathbf{I}))$. An update rule that exploits the *locality* of the problem structure may be obtained by considering that only $v_i$ needs to be updated in order to satisfy the force balance constraints $C_i = \{C_v, \forall v \in \mathcal{N}(v_i)\}$ at each time $t$. Assume $\Delta_{i,v}^t$ is the update value for $v_i$ obtained using the constraint $C_v$. This can be easily computed by substituting $v_i^t = v_i^{t-1} + \Delta_{i,v}^t$ in each $C_v \in C_i$. For instance, using $C_{v_i}$ we obtain,

$$\Delta_{i,v_i}^t = \frac{\alpha({v_i^{(2)}}^{t-1}) - \beta({v_i^{(4)}}^{t-1}) + \mathbf{f}(v_i^{t-1})}{(2\alpha + 6\beta)}$$

where ${v_i^{(k)}}^t$ is the $k$-th derivative at time $t$ at $v_i$. For simplicity, we have assumed $\alpha_i = \alpha, \beta_i = \beta, \forall i = 1, \ldots, n, \forall t$. We generate $|\mathcal{N}(v_i)|$ estimates for the update at $v_i$, one for each constraint $C_v \in C_i$. In order to combine these estimates into a single update, we need to *equalize* them. This may be achieved by a simple averaging operation but one may seek a more appropriate *equalizer* to obtain better performance. For results in this paper,

$$\Delta_i^t = \frac{\sum_{v\in\mathcal{N}(v_i)} \Delta_{i,v}^t}{|\mathcal{N}(v_i)|}, \quad \forall i = 1, \ldots, n$$

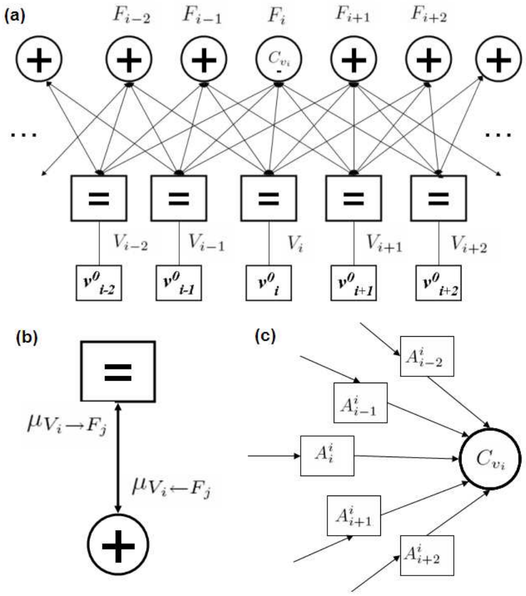


**Fig. 1**. Factor graph representation of Active contours.
(a) Bipartite graph AC FG encodes $\mathbf{A}$. Variable node $V_i$ is sqaure equals with initial condition $\mathbf{v}^0$. Composite function node $F_j$ is circle plus. (b). Bidirectional edge update (c). Structure of the composite plus node.

To achieve these updates systematically and across iterations, consider the FG given in Fig. 1. This bipartite graph encodes the structure of the matrix $\mathbf{A}$. Node $V_i$ corresponds to variable $v_i$ (*soft equals*) and the function node $F_j$ (*soft plus*) corresponds to the constraint $C_{v_j}$. There is an edge between $V_i$ and $F_j$ (i.e. $V_i \in \mathcal{N}_{FG}(F_j)$ and vice versa) whenever $v_j \in \mathcal{N}(v_i)$. At each time $t$, $V_i$ has a *belief* $b_i^t$ based on all input messages and at the end of the algorithm, we threshold $Th(b_i^{\xi(t)})$ to get the final values of $v_i, \forall i = 1, \ldots, n$. The computation on AC FG may be done using the Message passing schedule given in Algorithm 1. The stopping rule $\xi(.)$ may be $t \leq T$ or till the contour stabilizes. For the numerical setting discussed above, $\mu^t_{V_i \to F_j} = b_i^t = b_i^{t-1} + (\sum_{F_j \in \mathcal{N}_{FG}(V_i)} \mu_{F_j \to V_i})/|\mathcal{N}_{FG}(V_i)|$, $\mu^t_{F_j \to V_i} = \Delta^t_{i,v_j}$ and $v_i^{\xi(t)} = Th(b_i^{\xi(t)}) = b_i^{\xi(t)}$. Next we derive a Message passing formulation using Gaussian messages on FGs of linear models [5], [6]. To extend the above framework to a statistical setting, we let $v_i^t = (x_i^t, y_i^t)$ be a bivariate Gaussian r.v's, i.e. $V_i \sim \mathcal{N}_2(m_x, m_y, \sigma_x, \sigma_y, \rho)$. Note that the parameters $m_x, m_y$ denote the center of spatial certainty and $\sigma_x, \sigma_y$ denotes the spread along the image dimensions.

Control point updates in AC energy minimization may be seen as output from a linear system $\{X_t = AX_{t-1} + BU_t; Y_t = C_t X_t\}$ with $\mathbf{v}^t = X_t, U_t = 0, C_t = I \Rightarrow Y_t = \mathbf{v}^t$ and with initial conditions $\mathbf{v}^0$. We derive a novel Gaussian Message passing algorithm for AC by using the message update rules for soft plus and soft equals derived in [7] for a vector setting. For the sake of simplicity and space, we will only state the message update rules for the univariate Gaussian messages $\mathcal{N}(m, \sigma^2)$ in Fig. 2. This is to motivate the essential idea but our algorithm uses the messages in a generic setting that are elegantly derived in [5]. The messages passed along the edges of the AC FG and beliefs of variable nodes are parameters of bivariate Gaussian distribution. We threshold as $v_i^{\xi(t)} = (x_i^{\xi(t)}, y_i^{\xi}(t), Th(b_i^{\xi(t)}) = (m_x, m_y)$.

**Algorithm 1** MESSAGEPASSINGAC $[\mathbf{A}, \mathbf{v}^0, n = |\mathbf{v}^0|, \xi(.)]$

```
Construct AC FG (F ∪ V, A), Edges from A
Initialize params; b^0 from v^0;  t ← 0
while ξ(t) do
  (For all the equals gates)
  for all V_i : i = 1 to n do
    d_i ← |N_FG(V_i)|
    (Compute {μ_{V_i→F_j} : F_j ∈ N_FG(V_i)})
    for all j = 1 to d_i do
      μ_{V_i→F_j} = equals({μ_{F_k→V_i} : F_k ∈ N_FG(V_i)}\
                        μ_{F_j→V_i}, b_i^{t-1})
    end for
    (Update the belief)
    b_i^(t) ← B({μ_{V_i→F_k} : F_k ∈ N_FG(V_i)}, b_i^{t-1})
  end for
  (For all the plus gates)
  for all F_j, j = 1 to n do
    d_j ← |N_FG(F_j)|
    (Compute {μ_{F_j→V_i} : V_i ∈ N_FG(F_j)})
    for all i = 1 to d_j do
      μ_{F_j→V_i} = plus({μ_{V_k→F_j} : V_k ∈ N_FG(F_j)}\
                        μ_{V_i→F_j}, A^j, C_{v_j})
    end for
  end for
  EDGE UPDATE: Exchange messages
  t ⇐ t + 1
end while
RETURN {v_1, ..., v_n} ⇐ Threshold(b_1^{ξ(T)}, ..., b_n^{ξ(T)})
```

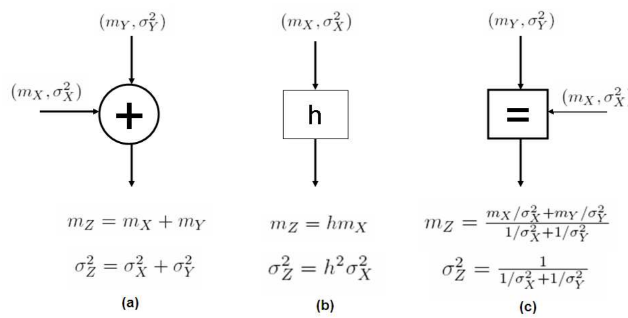


**Fig. 2**. Gaussian message update rules (univariate). (a) Soft plus (b) Product gate (c) Soft equals

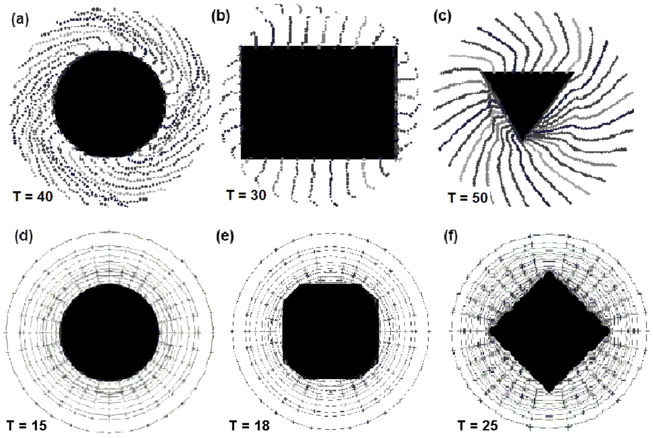


**Fig. 3**. Shape detection using Active contours (T is the number of iterations required for convergence): (a), (b), (c) using Dynamic programming (d), (e), (f) Message passing algorithm

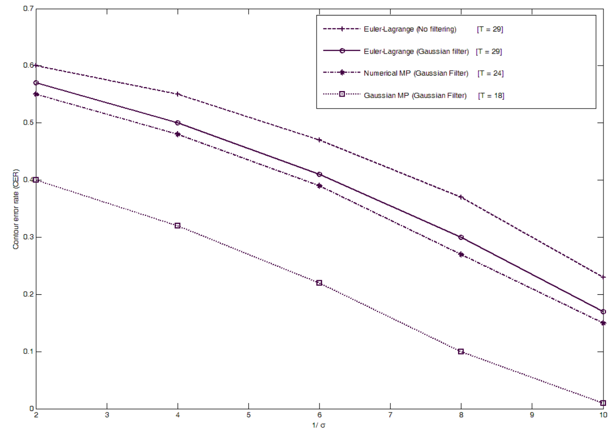


**Fig. 4**. Contour error rate (CER) for varying (Gaussian) noise levels $(1/\sigma)$ and Mean number of iterations for convergence (T). Contour size = 50. Compared algorithms are Euler-Lagrange (with and without Gaussian filtering), Numerical Message passing, Gaussian Message passing (with filtering)

## 4. SIMULATION RESULTS

We simulated shape detection using Active contours on randomly generated convex polygons with number of sides ranging from 3 to 50 using ACs of sizes $n, 32 \leq n \leq 100$. The images were grayscale with varying amount of Gaussian noise. Fig. 3(d), 3(e), 3(f) show convergence results from specific runs of the Message passing algorithm. We define the number of contour points which do not lie on any of the sides of the polygon (after the contour stabilizes) as *contour errors* and hence we have *Contour Error Rate* (CER) = *(number of contour errors)*$/n$. All four algorithms, namely Euler-Lagrange method (with and without Gaussian filtering), Numerical Message passing and Gaussian Message passing were rigourously tested for accuracy and computational complexity (number of iteration required for convergence). The results of the simulations are shown in Fig. 4. along with the average number of iterations to convergence. We observe that the performance of all four algorithms improves as the noise decreases but the Gaussian Message passing algorithm offers substantial gains in CER and number of iterations required for convergence even in presence of higher noise levels. As

## 5. CONCLUSIONS

In this paper we have formulated energy minimization in Active contours as an instance of Gaussian Message passing on Factor graphs to obtain substantial gains over existing numerical methods in terms of computational complexity, performance and robustness. This suggests that by making suitable statistical assumptions about the system, the existing numerical techniques may be extended to derive Message passing algorithms which exhibit improved computational complexity, numerical stability and robustness to noise.

The problem of ACs is a specific instance of energy minimization which in often is computationally intractable. This approach may also be extended to combine the Message passing algorithm for energy minimization with unconstrained optimization methods like gradient descent to produce powerful algorithms which are computationally tractable.

## 6. REFERENCES

[1] M Kass, A Witkin, and D Terzopoulos, “Snakes: Active contour models,” *International Journal of Computer Vision*, 1988.

[2] F R Kschischang, B J Frey, and H.-A Loeliger, “Factor graphs and the sum-product algorithm,” *IEEE Transactions on Information Theory*, 2001.

[3] H.-A. Loeliger, “An introduction to factor graphs,” *IEEE Signal Proc. Mag*, 2004.

[4] H.-A. Loeliger, “Some remarks on factor graphs,” *Proc. 3rd International Symposium on Turbo Codes and Related Topics*, 2003.

[5] H.-A. Loeliger, S Korl, J Hu, Q Guo, and Li Ping, “Gaussian message passing on linear models: an update,” *Proc. International Symposium on Turbo Codes and Related Topics*, 2006.

[6] G. D. Forney Jr, “Codes on graphs: normal realizations,” *IEEE Trans. Information Theory vol. 47*, 2001.

[7] H.-A. Loeliger, “Least squares and kalman filtering on forney graphs,” *Codes, Graphs, and Systems, (festschrift in honour of David Forney*, 2002.