Journal of Theoretical and Applied Information Technology
31st March 2015. Vol.73 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

BROKEN CHARACTER RESTORATION USING GRADIENT VECTOR FLOW AND BALLOON FORCE ALGORITHM

[1]QUSAY O. MOSA, [2]MOHAMMAD F. NASRUDIN

[1]Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, Malaysia
[2]Center for Artificial Intelligence Technology, Universiti Kebangsaan Malaysia, Malaysia
E-mail: [1]qusaymosa@yahoo.com, [2]mfn@ ukm.edu.my

ABSTRACT

The presence of a large number of broken characters in a digital document image represents the main problem in Optical Character Recognition (OCR) of historical documents. This problem still continues a challenge for recent OCR solutions. Broken character restoration from historical documents it is substantial because these documents contain important facts and leaving them leads to losing invaluable information. Gradient Vector Flow (GVF) snake has much more capture range than traditional snake therefore widely used in image segmentation. In this paper we used balloon with triangle steps to improve GVF snake to converge of deep concavity area and Restore broken characters using the improved GVF.

Keywords: *GVF, Snake Algorithm, Divergence, Deep Concavity, Balloon Force*

1. INTRODUCTION

Optical Character Recognition, usually referred to as OCR, is the process of converting the image obtained by scanning a text or a document into machine-editable format. Difference in font and sizes makes recognition task difficult if pre-processing, feature extraction and recognition are not robust. Accordingly, to get a good character recognition approach must be first remove the noise after reading binary image data, smooth the image for better recognition, extract features efficiently, train the system and classify patterns. There are several reasons led to degradation, including aging of the paper, incomplete printed characters, and blots of ink on the original document [1].

The presence of a large number of broken characters in a digital document image represents the main problem in Optical Character Recognition (OCR) of historical documents. This problem still continues a challenge for recent OCR solutions. Some of old historical documents are in bad condition and therefore their broken or degraded characters need to be restored or reconstructed. Broken character restoration from historical documents it is substantial because these documents contain important facts and leaving them leads to losing invaluable information [2]. Few research have been done in broken character restoration, the reason lies in lost information of broken characters by means of ink fading, inappropriate scanning, paper quality, exhausted copier and more other cases [3]. Some research reported to recognize broken characters by method of Hidden Markov Model (HMM) [4]. A technique has been proposed to recognize broken Hindi characters by using Multilayer feed-forward neural network [5]. On the other hand, Chain Method and template matching is used to Recognize Broken and Touching Persian characters [6]. Besides that, Dynamic Bayesian network has been used to recognize degraded characters [7], while optimized nearest neighbor to recognize and reconstruct broken handwritten digits [8].

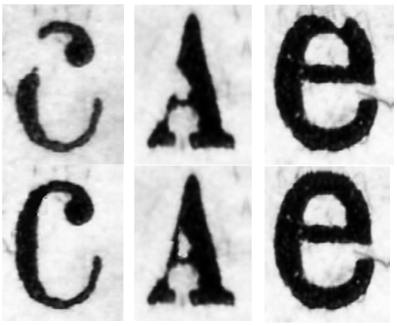

Figure 1: Broken Character examples

1.1 Active Contours

Journal of Theoretical and Applied Information Technology
31st March 2015. Vol.73 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Active contours or snakes are the most effective tools in image segmentation that have been widely used in this problem because they always produce sub-regions with smooth and continuous boundaries, on the contrary of the other segmentation techniques [8]. Active contours first introduced by Kass 1988), as a curve of points initialization close from the object. It iteratively segments image based on energy minimization until curves of object boundary is found. The curves are influenced by an external force which makes a curve flexible and prevent it from bending too much, and internal force that pushes the curve to object edges [9]. Cohen in 1989 introduced a balloon model to expand the snake's capture range by adding a second external force to push the curve outward or inward [10].

A new active contour model called the gradient vector flow (GVF) was proposed by Xu and Prince (1998). It uses a spatial diffusion of the gradient of an edge map of the image instead of directly using the image gradients as an external force [11]. The GVF snake was proposed to overcome the traditional snake's problem such as short capture range and convergence to concavity boundaries. The GVF snake has a drawback which is difficulty of convergence to deep concavity boundaries [12].

1.2 GVF Snake

In this section we will explain the GVF snake algorithm and a solution proposed to improve it. Let $X(s) = [x(s), y(s)], s \in [0,1]$ represents initial closed curve which will be formed by internal and external forces to convergence to edges. These forces are belongs to energy functions. Energy function of snake is mathematically represented using the following equation Kass *et al*. (1988).

$$E_{snake} = E_{internal} + E_{external} \quad (1)$$

The internal energy is defined as follows:

$$E_{internal} = \int_{s=0}^{1} \frac{1}{2}(\alpha|X'(s)|^2 + \beta|X''(s)|^2)ds \quad (2)$$

Where $X'(s)$ the first derivative of $X(s)$ represents the elastic energy which imposes the curve to shrink, $X''(s)$ is the second derivative of $X(s)$ which represents the bending energy which makes the curve tends to be like a thin plate. α and β are weighting parameters. By adjusting these weights, it will control the shrinking and thinning of the plate, besides that, by setting β to zero. The difference between GVF snake and traditional snake model lies in the external energies.

$$E_{external} = \int_{0}^{1} E_{image}\,(x(s))ds \quad (3)$$

For the GVF-snake, the external force $E\_external$ is replaced

VF-snake, the external force $E_{external}$ is replaced by gradient vector follow:

$$V(x,y) = \big(u(x,y), v(x,y)\big) \quad (4)$$

$V(x,y)$ can be obtained by minimizing the following energy function Xu and Prince(1988):

$$\varepsilon = \iint \mu(u_x{}^2 + u_y{}^2 + v_x{}^2 + v_y{}^2) + |\nabla f|^2|V - \nabla f|^2\, dxdy \quad (5)$$

Where: ∇f is the gradient of edge map, and μ is an adjusting parameter. In order to minimize the energy of Equation (5), it must satisfy the following Euler equation:

$$\mu\nabla^2 u - (u - f_x)\big(f_x^2 + f_y^2\big) = 0 \quad (6)$$

$$\mu\nabla^2 v - \big(v - f_y\big)\big(f_x^2 + f_y^2\big) = 0 \quad (7)$$

Where ∇^2 is the Laplacian operator, in homogenous regions both f_x and f_y are zero.

1.3 Deep Concavity Problem

GVF snake has much more capture range than traditional snake therefore widely used in image segmentation [13]. GVF has drawback in capture deep concavity object like "U" in some image and this happened in the process of snake approach to the object, control points move to the place has the lesser energy. It is not all pixels with local minimum force in the true boundary, therefore snake unable to converge to correct boundaries. The idea is to overcome the local minimum of some control points to make all pixels of active contour capture the right boundary.

The main objectives of this paper are:

i. Improve the GVF snake algorithm: by using balloon force in area determined by triangle steps (explained above) to converge of deep concavity area.
ii. Restore broken characters using the improved GVF and save time consumed in convergence computation.

2. MATERIALS AND METHODS

We propose a method to improve the GVF for broken characters restoration, as the following:

a) Calculate the GVF field as mentioned in Equations (1), (2) and (5).

Journal of Theoretical and Applied Information Technology
31st March 2015. Vol.73 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

b) Calculate the divergence to decide which points are not converged to the edge and remove them.

$$\text{Let } F(x,y) = P(x,y)i + Q(x,y)j \quad (8)$$

Be the GVF field of the image, where $P$and Q are the horizontal and vertical axis vectors respectively, and i, j are pixel coordinates.
Divergence is calculated by the equation:

$$div\ F = \frac{\partial P}{\partial x} + \frac{\partial Q}{\partial y} \quad (9)$$

From the divergence field, low values belong to the object boundaries, while large values to those areas which are far from the boundaries. Setting the threshold value θ (appropriately chosen threshold because value of θ is delimit the deep concavity area) to determine which values of $divF$with coordinate (i,j) reach the boundary when:

$$divF(i,j) < \theta \quad (10)$$

Therefore each point not satisfies (10) will be removed from the snake.
c) Triangle step: we will determine three snaxels (points which is form the snake curve) in the deep concavities. First we consider consecutive (cut-off) snaxels $A(x_1, y_1)$ and $B(x_2, y_2)$ for whose distance d is defined:

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (11)$$

Let the equation of the line that passes through snaxels A and B be:

$$y = ax + b \quad (12)$$

Line (12) divides the Cartesian plane into two half planes. New line is vertical to line (12) and goes through the middle of the line segment AB. To force the snake into the concavity, we then add a new snaxel C lying on line (12). From line (12) the parameter C adjusts the depth where the new snaxel is defined. The three snaxels A, B and C form a triangle.
d) With triangle points (A, B, C) we used Balloon technique as a second external force to convergence the line into deep boundary until point C.
e) Then calculate a divergence and check all snaxels of snake according to equation (10).
f) As shown below the example of deep concavity (character "U") and processing of triangle step with Balloon force.

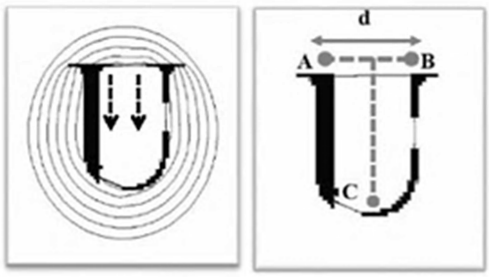

Figure 2: (a) Deep Concavity (b) Triangle Step

2.1 Performance Evaluation

In this study, we use two types of metrics to evaluate the performance of traditional and improved GVF algorithms:
a) Hausdorff Distance (HD): as the way to evaluate performance. It is a measurement tool that explains the degree of similarity between two sets of points or images. HD is the maximum distance from a point set to the nearest point in other point set.

If $A = (a_1, a_2, \ldots, a_n)$ and $B = (b_1, b_2, \ldots, b_n)$ are two sets and HD is represented in following equations:

$$H(A,B) = \max[h(A,B), h(B,A)] \quad (13)$$

Where:

$$h(A,B) = max_{a\in A}\{min_{b\in B}\|a-b\|\} \quad (14)$$

$$h(B,A) = max_{b\in B}\{min_{b\in B}\|b-a\|\} \quad (15)$$

When two images are identical, the HD will be zero. So as will show in table1 and table 2, HD is calculated for broken images of Latin alphabet (A-Z). When use GVF snake alone which suffers from deep concavity for example character "H" HD value is 191.1345 while when use GVF improve HD value is 6.9282 with size of 32×32. Also with table2 when use HD to evaluate broken images of MNIST database. The MNIST database contains 60,000 digits ranging from 0 to 9 for training the digit recognition system, and another 10,000 digits as test data. Each digit is normalized and centered in a gray-level image with size 28 × 28. Some examples are shown in Figure 3.

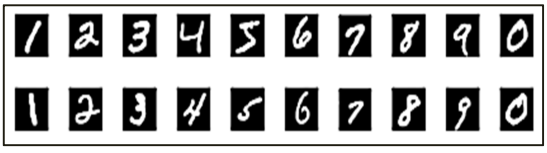

Figure 3: Examples of MNIST Data Set

Journal of Theoretical and Applied Information Technology
31st March 2015. Vol.73 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table1: HD between GVF Alone and Improved GVF

character	GVF	GVF Improved	
		64 x 64 pixels	32x 32 pixels
A	161.5952	151.3819	39.1535
B	196.7438	166.9642	27.6586
C	161.6146	122.0950	27.7669
D	205.5505	157.0115	29.5466
E	155.4512	135.3332	6.7082
F	151.3505	123.7437	8.6603
G	175.2978	154.5998	28.4077
H	191.1348	122.0145	6.9282
I	113.1702	100.0013	3.0000
J	114.1855	82.9077	28.6182
K	161.7868	149.9735	27.6586
L	120.2520	72.6792	14.1774
M	228.1553	176.2504	62.6658
N	197.2872	179.1435	47.0532
O	200.6883	162.5032	39.3065
P	157.7898	150.3040	29.0345
Q	202.0967	192.8451	13.6382
R	198.9340	155.0533	28.8271
S	160.3438	111.9165	11.0905
T	143.4684	137.3574	49.1833
U	174.7361	138.4035	56.0179
V	136.4806	105.7382	38.6523
W	198.9759	168.7516	46.9255
X	153.1587	143.2014	28.6705
Y	139.9883	98.5015	31.2730
Z	161.0201	134.2415	27.8747
Average	161.3075	142.8117	33.5141

b) Convergence/ Running time: the time taken to compute the segmentation including the number of iteration needed to converge is shown as in table3, which represents GVF processing time to restore broken characters like (A, B, D, O) and recognized in few milliseconds. While GVF unable to converge to broken characters like (C, E, F, H, S, U) and stops in beginning of deep concavity area that means local minimum and considering this minimizing energy therefore use GVF improved to restored these character which signed by (*) in table3. We test 100 samples for each data set (0-9) of training digits. Table 2 contains samples of test MNIST dataset by HD:

Table2: HD on MNIST Dataset

MNIST characters		Hausdorff Distance		
			GVF improved	
sets	sample	GVF	64 x 64 pxls	32 x 32 pxls
0_set	Zero.1	48.0521	83.2286	35.1710
	Zero.2	17.1756	29.7489	16.1864
	Zero.3	58.7026	101.6760	33.0606
	Zero.4	32	55.4256	23.5372
1_set	One.1	8.3542	10.5826	6.4031
	One.2	46.8935	44.8330	18.4391
	One.3	13.4907	19.9750	7
	One.4	66.1967	19.7484	8
2_set	Two.1	32.2645	55.8838	16.1245
	Two.2	30.7246	72.0833	19.3649
	Two3	7.1414	27.6586	15.9687
	Two.4	36.7287	62.4500	29.5127
3_set	Three.1	26.7395	46.3141	19.9249
	Three.2	19.0788	33.0454	11
	Three.3	30.9354	53.5817	6.5574
	Three.4	16	27.7128	8.4853
4_set	Four.1	53.3854	92.4662	36.2767
	Four.2	23.3024	40.3609	20.6640
	Four.3	39.5348	68.4763	32.4962
	Four.4	5.3852	9.3274	4.4721
5_set	Five.1	43.6119	75.5381	22.6274
	Five.2	46.0760	79.8060	28.3019
	Five.3	35.7911	61.9919	25.0998
	Five.4	25.8650	44.7996	22.8910
6_set	Six.1	28.0891	48.6518	21.9545
	Six.2	27.6586	47.9062	16.6733
	Six.3	4	6.9282	7.9373
	Six.4	38.0132	65.8407	30.2655
7_set	Seven.1	25.1595	40.0999	20.9762
	Seven.2	32.6956	56.6304	27.9643
	Seven.3	16.3095	28.2489	16.2788
	Seven.4	32.5115	56.3116	22.8254
8_set	Eight.1	23.3666	40.4722	21.2838
	Eight.2	22.8035	39.4968	17.7764
	Eight.3	23.5797	40.8412	19.3649
	Eight.4	22.6053	39.1535	15.8114
9_set	Nine.1	32.0780	56.3383	16.0935
	Nine.2	38.3536	66.3626	28.9310
	Nine.3	22.6053	39.1152	16
	Nine.4	22.8035	39.4968	16.2481

Table 3: Running Time Evaluation on Broken Characters

Character	No. of iterations	Time/millsec
A	75	9.00
B	40	5.00
C*	90	10.23
D	35	4.38
E*	90	10.23
F*	95	10.85
G	50	6.16
H*	105	11.83

Journal of Theoretical and Applied Information Technology
31st March 2015. Vol.73 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

I	70	8.05
J	80	9.09
K	100	11.26
L	90	10.23
M	110	12.36
N	115	12.81
O	40	5.07
P	70	8.07
Q	55	6.50
R	75	9.00
S*	85	9.69
T	70	8.07
U*	170	18.68
V	120	13.68
W	145	16.19
X	65	7.56
Y	65	7.56
Z	105	11.75
Average	**85**	**9.74**

3. RESULTS AND DISCUSSIN

The proposed method is tested on broken character images of Latin alphabet taken from ISO basic Latin alphabet and images from the MNIST database. It gives a better result compared with GVF alone. All computation is done under the environment of DELL INSPIRON N5110 computer. CPU 2.4GHz and RAM is 4 GB. The operation system is Windows7 Ultimate and the programming tool is MATLAB 7.10.0 (R2010a). Vextractor 6.42 software is used to convert raster image to vector image.

To test the performance of the proposed method, below is the explanation of steps to restore broken characters:

a) Take a character image size of 64×64 (e.g. digit 3) and run procedure of segmentation by the GVF snake algorithm with parameters setting of $\alpha = 0.01$, $\beta = 0$, $\gamma = 1$ and $\mu = 0.6$. The result does not converge to deep concavity of boundaries as shown in figure 4(a).

b) Run GVF improved method on the same digit (3) and same parameters by calculates divergence according to equation (8) and each point not satisfies θ ($\theta = 0.03$) it will be removed from the snake. Then process the triangle steps for setting three points represent the line across the deep area and the depth line perpendicular on across line.

c) Process balloon algorithm as a second external force to expand snake in deep area determined by triangle steps and converge to deep concavity boundaries.

The result as shown in figure 4(b) demonstrates the proposed method has overcome the deep concavity problem.

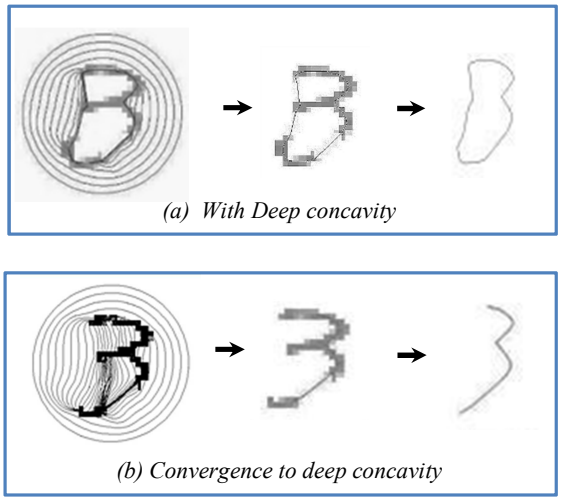

Figure 4: MNIST Dataset Sample (digit "3")

d) Continue to test all broken characters images using the improved GVF.

e) Run vextractor software on the not-broken original images of characters to converting it from raster to vector images. We use the vector image as the ground truth.

g) Calculate Hausdorff Distance factor to evaluate the matching rate, first: between images resulted from the GVF alone and the vector image we got it from vextractor software, second: between the images we got from new GVF improved and vector images as explained in figure 4.

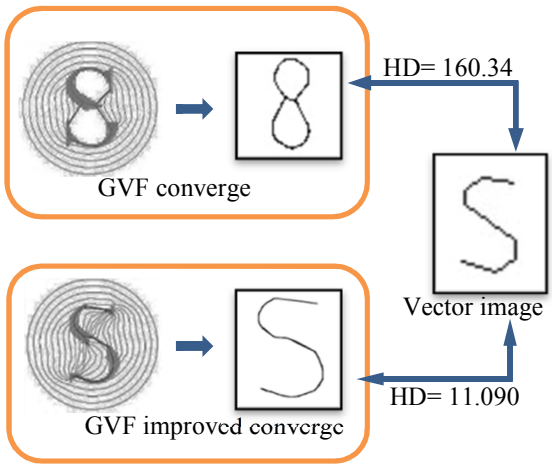

Figure 5: HD with Vector Image

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

4. CONCLUSION

Experimental results from this study prove the effectiveness of the improved GVF to restore broken characters compared to the GVF alone. It is done by combining the GVF field and the balloon forces. Besides connecting the broken images, another advantage of the improved GVF is that it can capture deep concavity boundaries as well.

REFERENCES

[1] Subramaniam Venkatraman, "Degradation Specific Ocr", Master thesis of Science in Electrical Engineering, Boise State University, 2010.

[2] Manas Y., Manas R. Nayak, Subhagata C., "Recognition and Classification of Broken Characters using Feed Forward Neural Network to Enhance an OCR Solution", International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 1, Issue 8,2012.

[3] M. K. Shukla, Dr. H. Banka," A Study of Different Kinds of Degradation in Printed Bengali Script". I JACEA Vol.2, 2012, pp. 143-151.

[4] C. B. Bose, S-S Kuo, "Connected and degraded text recognition using Hidden Markov Model". Pattern Recognition, Vol. 27, No. 10, 1994, pp. 1345-1363.

[5] Yetirajam M., Nayak M.R., Chattopadhyay S, "Recognition and Classification of Broken Characters using Feed Forward Neural Network to Enhance an OCR Solution", I J AR in Computer Engineering & Technology, Vol. 1, No. 8,2012, pp. 11-15.

[6] A.H. Pilevar, M.T. Pilevar, "Broken and Touching Characters Recognition in Persian Text Documents", world Applied Sciences Journal Vol. 13, No. 6, 2011, pp. 1459-1464.

[7] L. L. Sulem, M. Sigelle,"Recognition of degraded characters using dynamic Bayesian networks", Journal of pattern Recognition, vol. 41, 2008, pp. 3092-3103.

[8] D. Yu, H. Yan, "Reconstruction of broken handwritten digits based on structural morphological features", The Journal of Pattern Recognition Society, 2001,vol.34, pp. 235-254.

[9] Rajeshwar Dass, Priyanka, Swapna Devi," Image Segmentation Techniques", IJECT Vol. 3, Issue 1, 2012.

[10] M. Kass, A. Witkin, D. Terzopoulos, "Snakes: active contour model"s, 749 International Journal of Computer Vision, 1988, 1 (4) 321–331.

[11] L. Cohen, "On active contour models and balloons". Computer Vision, Graphics, and Image Processing: Image Understanding, 1989, 53(2):211–218.

[12] C. Xu and J. Prince, "Snakes, shapes, and gradient vector flow", *IEEE Transactions on Images Processing,*1998,7(3):359– 369.

[13] L. He, Z. Peng, B. Everding et al., "A comparative study of deformable contour methods on medical image segmentation", Image and Vision Computing, vol. 26, no. 2,2008, pp. 141–163.

[14] Jinyong Cheng, Xiaoyun Sun ," Medical Image Segmentation with Improved Gradient Vector Flow", Research Journal of Applied Sciences, Engineering and Technology 4(20),2012,: 3951-3957.