Image Segmentation Based on Pulse Coupled Neural Network

Dansong Cheng Wei Zhao   Xianglong Tang   Jiafeng Liu
Department of Computer Science and Engineering, Harbin Institute of Technology
Harbin,150001, China
e-mail: *cdsinhit@hit.edu.cn*

Abstract: This paper introduces an approach for image segmentation by using pulse coupled neural network (PCNN), based on the phenomena of synchronous pulse bursts in the animal visual cortexes. The neighborhoods considering magnitude and distribution of the pixels are modeled into a factor to control the linking, further, control the internal activity. Segmentation criteria based on majority rule controls the process, and threshold adjustment within one iterative ensures the integrated result. Experiments on several types of image are implemented with the proposed method and the experimental results comparing with classical methods demonstrate its validity.

Keywords: Pulse-Coupled   Neural Network (PCNN); Image Segmentation; square neighborhood

1. Introduction

Pulse coupled neural networks (PCNN) were introduced as a simple model for the cortical neurons in the visual area of the cat's brain. Important research in the 80's and 90's led to the establishment of a general model for PCNN [1, 2]. Such models are proved to be highly applicable in the field of image processing, a series of procedures being developed for contour detection and especially image segmentation [3-10]. Taking PCNN process into consideration, the performance depends on the proper selection of the parameters. Although some researches tried to find criterions to select adaptively parameters based on intensity statistical properties[11], image entropy[12] or other image quality indictor[13], holistic adjustment are insufficient to reach an optimal standoff between false negative and false positive. We propose an extended PCNN using fast linking, and make some improvements to extend PCNN to work effectively. Some parameters are adjusted based on its neighborhood for each pixel, and the neighborhood is modeled accurately to a ratio. We apply this PCNN to several types of image under various conditions of illuminations and demonstrate the effectiveness of this model through experiments.

In the second section of this paper, the PCNN's basic model and the fast-linking method are introduced. In the third section, the new approach for image segmentation based on fast linking is brought forward. In the fourth section, results of segmentation based on the proposed PCNN model are shown.

2. THE PULSE-COUPLED NEURAL NETWORK

2.1. PCNN neuron model

The Pulse-Coupled Neural Nets (PCNN) algorithm is based on the neurophysiologic models evolving from studies of small mammals. Shown in Fig.1, the PCNN will receive both stimulus by feeding and also inhibitory linking. These are combined in an internal activation system. Which accumulates the signals

until it exceeds a dynamic threshold, resulting in an output. This alters the threshold as well as linking and feeding neurons, as will be described below. The PCNN produces a temporal series of outputs. Depending on time as well as the parameters, this dynamic output contains information, which makes it possible to detect edges, do segmentation, identify textures and perform other feature extractions. The PCNN can operate on different types of data since it is very generic to its nature. The algorithm is performed by continual iterations of the input and the output using the following equations: F channel and L channel combine in a second order fashion to cerate the total internal activity U, which is then compared to the dynamic neuromime threshold θ to create the output Y. If the internal activity U is large enough, then the neuron will generate a pulse. α_F, α_L and α_θ are the decay constants of the PCNN neuron. V_F, V_L and V_θ are the magnitude scaling terms. M and W are the synaptic weigh strengths defining the feeding receptive field and linking receptive field, respectively. The constant β is the linking strength.
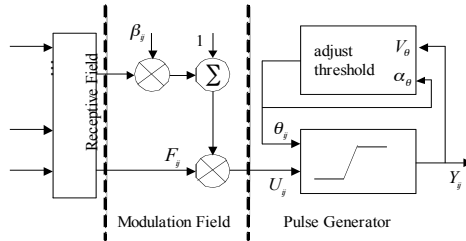

Figure 1. PCN neuron

$$F_j[n] = e^{-\alpha_F} F_j[n-1] + V_F \sum MY[n-1] + S_j \quad (1)$$

$$L_j[n] = e^{-\alpha_L} L_j[n-1] + V_L \sum WY[n-1] \quad (2)$$

$$U_j[n] = F_j[n](1 + \beta L_j[n]) \quad (3)$$

$$\theta_j[n] = e^{-\alpha_\theta} \theta_j[n-1] + V_\theta Y_j[n] \quad (4)$$

$$Y_j[n] = \text{Step}(U_j[n] - \theta_j[n]) = \begin{cases} 1 & U_j[n] > \theta_j[n] \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

The image processing PCNN used in our experiments is a single layer two-dimensional array of laterally linked neurons and all neurons are identical. The number of neurons in the network is equal to the number of pixels in the input image. There exists a one-to-one correspondence between the image pixels and neurons.

2.2. Fast Linking Method

In order to process actual images under various conditions of illuminations, we apply PCNN with "fast linking"[3], after the first signal is input, calculate all the output, and then refresh the linking territory. At last, the internal state is calculated, the output is decided. During the calculating process, if one of the neurons is changed, the linking territory is changed correspond. The calculation will continue until all the outputs are unchanged. Such cycle process is called one iterative. During this process, in order to keep the input unchanged, the linking territory will change constantly. The input wave transmit the data after one iteration is finished, while linking territory wave send information to all the elements of image during this iterative. This method is called Fastlinking. It can decrease the effect of timing quantification. The flashings in original model are all separated because the time delay of the linking territory. While adopting the Fastlinking model, the neuron can be flashed in one territory. Shown in Fig 2.

In the model of PCNN, the linking coefficient β plays an important role. The larger β is, the further distance is transmitted. It can be obviously seen in the Fastlinking model. Figure 3 shows the segmentations of different β. It also can be seen from Figure3 (b) and (c), for the connected region, the value of β influences the amount of neurons.
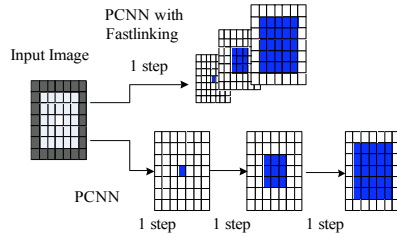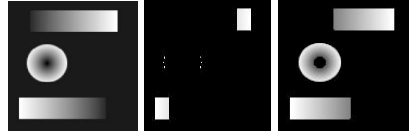
Figure 2. Fastlinking model

In the model of PCNN, the linking coefficient β plays an important role. The larger β is, the further distance is transmitted. It can be obviously seen in the Fastlinking model. Figure 3 shows the segmentations of different β. It also can be seen from Figure3 (b) and (c), for the connected region, the value of β influences the amount of neurons.

(a) Original image (b) $\beta = 0.2$ (c) $\beta = 5$

Figure 3. Segmentations with different β

3 Neighborhood inspiring factor

To illuminate the model clearly, we define an *image* **I** as a pair (l, *I*) consisting of a finite set l of *pixels* (points in *I*), and a mapping *I* that assigns to each pixel a *pixel value I(p)* in some arbitrary value space.

3.1 Neighborhood Coherence Factor

In most image-processing applications, it is justified to group the nearby pixels together since they are more likely to belong to the same region. Generally speaking, this cue is characterized by their spatial distance on the image plane. The formal mathematical description is $d(p,q) \le \tilde{n}, \forall p,q \in \mathsf{l}$, where $d(p,q)$ denotes the Euclidean distance and ρ is a specified constant [14]. For a fixed central pixel p, that $N_p = \{q : d(p,q) \le \rho, \forall p,q \in \mathsf{l}\}$ represents a

neighborhood of the pixel p. Here we apply the definition of the 1/r kernel of PCNN as the neighborhood. The formal mathematical description same as in [15] is

$$N_p = \left\{ q(x_q,y_q) : \left| x_q - x_p \right| \le \rho, \left| y_q - y_p \right| \le \rho, q \in \mathsf{I} \right\}$$

Suppose that the square neighborhood of each pixel is given, there are predicatively the pixels in the set

$$\Omega_p = \left\{ q \in N_p : d(I(p),I(q)) \le \varepsilon \right\}$$

(6)

and the pixels in the set

$$\Omega'_p = \left\{ q \in N_p : d(I(p),I(q)) > \varepsilon \right\}$$

(7)

for an arbitrary threshold $\varepsilon \ge 0$, where $\Omega_p \cup \Omega'_p = N_p$ and $d(I(p),I(q)) = \left| I(p) - I(q) \right|$ (8)

is a *pixel-value difference measure*. Here, a pixel in Ω_p is a ε *-similar neighbor* (εSN) of p. If the intensity difference between a central pixel p and its neighboring pixels is very small (below a threshold ε), it is conceivable that pixel p will be an interior point of one interested region and could be taken as a seed pixel used to grow the region. In contrast, if the intensity difference between the central pixel p and its neighboring pixels is rather high (greater than a threshold ε), the p would be out of one planar surface and lose the growing ability. Intuitively, if the number of neighboring pixels with the intensity values approaching to the central pixel's exceeds those far away from the central pixel's, we say that the central pixel could also be taken as a seed because the similar or coherent pixels within its neighborhood are in the ascendant. Motivated by this phenomenological explanation of image formation, we now define one neighborhood coherence factor (NCF) as follows:

$$\mathrm{NCF}(p) = \frac{\left| \Omega_p \right|}{\left| N_p \right|}$$

(9)

where $|\bullet|$ refers to the cardinality of a set, i.e., the number of elements in a set. It is defined to be the ratio of the number of pixels having the similar intensity with p to the number of pixels having the distinct intensity with p. Obviously, this value is quite discrepant for different pixels. When $|\Omega_p| \geq |\Omega'_p|$, $\text{NCF}(p) \geq 0.5$. In such a situation, p is similar to most of its neighboring pixels. When $|\Omega_p| < |\Omega'_p|$, $\text{NCF}(p) < 0.5$. That implies few of its neighboring pixels are similar to p. Therefore, we can say that $\text{NCF}(p)$ is actually an ε -similarity between p and its surrounding pixels with respect to ρ . In [14], $\text{NCF}(p) \geq 0.5$ implies the similar pixels will predominate over a handful of discontinuity pixels with sharp varying intensity values. Similarly, when $\text{NCF}(p) < 0.5$ the discontinuity pixels with sharp varying intensity values will predominate over the minority similar pixels.

3.2 Pixels captured

In PCNN model, the pixels with higher intensity than threshold pulse firstly. Here, we denote the set of these pixels as SEEDs:

$$\text{SEED} = \left\{ p : I(p) \geq T, p \in I \right\}$$

(10)

Where T is the threshold. These pulsed pixels will capture their neighbor pixels through the linking field. The linking is determined by W, M and β . In the classical fastlinking models (J.L. Johnson), the linking is generally assumed to be a constant. If 1/r kernel is applied, W and M are both defined as kernel matrix K_r :

$$K\_r(k,l) = \begin{cases} 1 & k = l = C_r \\ 1/\text{sqr}((k - C_r)^2 + (l - C_r)^2) & otherwise \end{cases}$$

where $C_r = \text{round}(r/2)$. β is also set a constant. That means the pixels with stimulus below one certain value will not be captured whether they are belong to the

object or not. So the results of these models, as shown in figure 3, shows that all the pixels are not segmented in one iterative.

If p is a SEED in iterative i, q is one of its neighbors, $S_q(i)$ is the feeding of q, $T(i)$ is the threshold, the linking is :
$L_q(i) = V_L \sum WY[i-1]$, Then its internal activity is $U_q(i) = S_q(i)(1 + \beta L_q(i))$. If q will be captured, the following equation should be satisfied:

$$U_q(i) = S_q(i)(1 + \beta L_q(i)) \geq T(i)$$

(11)

That means if $S_q(i) \geq T(i)/(1 + \beta L_q(i))$, q will be captured even though there is single pixel pulsed in its ρ -neighborhood.

With the analysis detailed in section 3.1 and 3.2, when $\text{NCF}(p) \geq 0.5$, the p is usually deemed to be ε -similar with its neighbors, i.e., the intensities of majority pixels surrounding vary slowly; when $\text{NCF}(p) < 0.5$, p is distinct from its neighbors, i.e., the intensities of majority pixels surrounding vary sharply. Further, the pixels with $\text{NCF}(p) \geq 0.5$, together with its nearest neighbors, would delineate all or part of an object with a high probability. Alternatively, the pixels with $\text{NCF}(p) < 0.5$ always lie between two different objects which may be the image border or the noise and some shadow boundaries.

Now, we define square neighborhood of q
$$\Omega(q,p) = \left\{ s \in N_q : d(I(s),I(p)) \leq \varepsilon \right\} \quad \text{and}$$
$$\Omega'(q,p) = \left\{ s \in N_q : d(I(s),I(p)) > \varepsilon \right\}$$
then modify the *neighborhood coherence factor* of q:

$$\text{NCF}(q,p) = \frac{|\Omega(q,p)|}{|N_p|}$$

Considering the influence of the distribution of εSN s, the limit of the partition is shifted to 0.6. Videlicet, the pixel q with $\text{NCF}(q,p) \geq 0.6$ should be

captured. If $\mathrm{NCF}(q,p) < 0.6$, considering the pixels in corners of the objects, q is treated in two ways:1) if $\mathrm{NCF}(q,p) \le 0.3$, q will not be captured; 2) if $0.3 < \mathrm{NCF}(q,p) < 0.6$, this implies it has three $\varepsilon\mathrm{SNs}$. Whether it is captured or not depends on the distribution of its $\varepsilon\mathrm{SNs}$. If the $\varepsilon\mathrm{SNs}$ locate close, as P_2 in figure 4, it should be captured. Alternatively, if its $\varepsilon\mathrm{SN}$ s distribute sporadically, as P_3 in figure 4, it should not be captured.
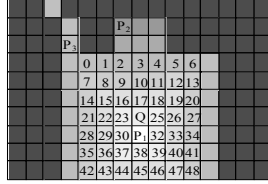
Figure 4 Illustration of the neighborhood

The measurement of the spatial dispersion of the $\varepsilon\mathrm{SN}$ s can be carried out with their *spatial dispersion factor* (SDF):

$$SDF(q) = \frac{1}{SDR1(q)} \cdot \frac{1}{SDR2(q)}$$

(12)

Where $SDR1(q)$ and $SDR2(q)$ are named as *space distance ratio* (SDR):

$$SDR1(q) = \frac{\sum_k ave(d(\varepsilon SN_k)) / (|N_P(1)| \cdot (|N_P(1)| - 1))}{\sum_{i=1}^{|N_p|} ave(d(s_i)) / |N_p|}$$

(13)

where $ave(d(s_i)) = \sum_{j=1}^{|N_p|} d(s_i, s_j) / |N_p|$ is the average distance between s_i and other neighbors, which is anyone in N_p;

$ave(d(\varepsilon SN_k)) = \sum_{l=1}^{|N_p(1)|} d(\varepsilon SN_k, \varepsilon SN_l) / (|N_p(1)| \cdot (|N_p(1)| - 1))$ is the average distance between εSN_k and other $\varepsilon\mathrm{SN}$ s, which is anyone in $\varepsilon\mathrm{SN}$ set, named as $N_p(1)$; $|\bullet|$ is same as in Eq.(9).

$$SDR2(q) = \frac{\sum_k d(\varepsilon SN_k, q) / |N_P(1)|}{\sum_{i=1}^{p} d(s_i, q) / |N_p|}$$

(14)

We can discover that SDR1 is competent for measurement of dispersion of $\varepsilon\mathrm{SN}$ s, and SDR2 for distance to q.

Some SDFs corresponding to certain neighborhoods are shown in table 1. Obviously, if $\mathrm{SDF}(q) \ge 1$, the $\varepsilon\mathrm{SN}$ s of q distribute closely and near to q. Alternatively, the pixels with $\mathrm{SDF}(q) < 1$, the $\varepsilon\mathrm{SN}$ s of q distribute sporadically and away from q.

Now, we define *neighborhood inspiring factor* (NIF):

$$NIF(q) = \begin{cases} NCF(q,p) / 0.3 & NCF(q,p) \ge 0.6 \\ SDF(q) & 0.3 < NCF(q,p) < 0.6 \\ 0 & NCF(q,p) \le 0.3 \end{cases}$$

(15)

With the analysis detailed above, $\mathrm{NIF}(q)$ is more comprehensive evaluation of pixels neighborhood. Table 1 shows the values of NCF, SDF and NIF corresponding to certain neighborhoods. $\varepsilon\mathrm{SN}$ indexes in the last row are according to Figure4.

When $\mathrm{NIF}(q) \ge 1$, q is ε-similar with its neighbors. If q is one of the ε-neighbors of a pulsed pixel, it is expected to be captured. The pixels with $\mathrm{NIF}(q) < 1$ always lie between two different objects which may be the image border or the noise and some shadow boundaries. They are expected not to be captured even if they are the ε-neighbors of pulsed pixels.

We modify the linking of the fastlinking model within one iterative

$$L_q(i) = NIF(q)V_L \sum WY[i-1] \qquad (16)$$

causing the change of the internal activity U. If we define

$$\varepsilon = |S_q(i) - T(i)| \qquad (17)$$

Table 1 NCF, SDF and NIF corresponding to certain neighborhoods

ρ	εSN	NCF	SDF	NIF
	$N_R(1,:)$	0.375	1.1599	1.1599
	$N_R(1,1:2) \cup N_R(2,1)$	0.375	1.5238	1.5238
3	$N_R(1,2) \cup N_R(3,1) \cup N_R(3,3)$	0.375	0.7345	0.7345
	$N_R(1,:) \cup N_R(2,1)$	0.5	1.1340	1.1340
	$N_R(1,1:2) \cup N_R(3,1) \cup N_R(3,3)$	0.5	0.7345	0.7345
	$N_R(1:4,1:4)$	0.3125	1.9362	1.9362
	$N_R(1:3,:)$	0.4375	1.2754	1.2754
	$N_R(1:3,:) \cup N_R(1:6,1:2)$	0.5417	1.1377	1.1377
7	$N_R(1:2,1:3) \cup N_R(1:2,5:7) \cup N_R(6:7,1:3) \cup N_R(6:7,5:7)$	0.5	0.6918	0.6918
	$N_R(1:2,:) \cup N_R(6:7,:)$	0.5833	0.7441	0.7441
	$N_R(1:3,:) \cup N_R(7,:)$	0.5833	0.8497	0.8497
	$N_R(1,2,4,5,14,16,19,23,25,26,27,28,35,36,42,43,44,45,46,47,48)$	0.4375	0.8378	0.8378

If q is ε-similar with a pulsed pixel p, and if $\text{NIF}(q) \geq 1$, according to eq.(11), $U_q \geq T(i)$, q will be captured by p. the lager $\text{NIF}(q)$ is, the stronger linking is. Alternatively, if $\text{NIF}(q) < 1$, $U_q < T(i)$, q will not be captured. That means q will be captured only if it is one ε-neighbors of some pulsed pixels, at the same time, it is ε-similar with its neighbors.

To ensure the "waves" move to the edge of the objects, we reduce the threshold by step $\varepsilon/255$ within one iterative. The threshold reduced is just for the neighborhoods of the pulsed pixels. In fact, ε need not be restricted by eq(17). A small offset may cause little difference in results. This will be discussed later.
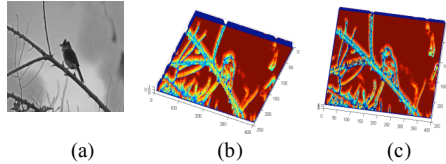

(a)   (b)   (c)

Figure 5 Illustration of NIF : (a) original image;(b) NCF values of image pixels ($\rho = 7$, $\varepsilon = 10$);(c) NIF of image pixels

4. precise segmentation using PCNN

4.1 *neighborhood inspiring PCNN (NIPCNN)*

Let the image after each iterative save in matrix T_{save}. The detailed steps are as followed:

Step1: Initialization

- Let the unitary image greyhound value as the impulse signal S_{ij};
- Initialize the parameter of the net;
- Set the maximum iterative times $n_{max} = 30$.

Step2: Let iterative variable $n = 1$

Step3: Fastlinking processing: t is the iterative variable

a) Let $t_{max}=20$; $t=1$;

b) Let the adjusted threshold $T_a(t)=T(n)$;

c) The first iterative $t=1$, from (1)~(5),calculate each PCNN internal and output part, then put the result in $Y_{ij}^{(t)}(n)$;

d) Find out the pixels pulsed in this sub-iterative, and calculate *NIF* of their neighbors;

e) Calculate the linking of these neighbors with Eq.(16);

f) Calculate the internal activity of these neighbors and compare with $T_a(t)$, then affiliate the output into $Y_{ij}^{(t)}(n)$.

g) $T_a(t) = T_a(t) - N(\varepsilon)$, where $N(\varepsilon)$ is the normalization of ε

h) $t=t+1$;

i) If $t<t_{max}$ then move to step c); otherwise output $Y_{ij}(n)$.

Step4: $n=n+1$. $T_{save}= Y_{ij}(n)$.

Step5: If n < nmax, then move to step 3; otherwise end and output Tsave, which is the result of segmentation.

4.2 Parameter Sensitivity to Segmentation Quality

As analysis in section 2.2, the effect of fastlinking model depends on the parameters. Appropriate parameters are determinant to the correct segmentation. Thus, adjustments for parameters according to each image ensure accurate result. Confessedly, it's a fussy task. Relatively, parameters have less influence on the effect of NIPCNN. Besides the parameters of the fastlinking, one new parameter, ε, is adopted into NIPCNN. The parameters inherited from fastlinking model are set same for all the images. r is set a constant, r =7 in this paper. As mentioned before, ε

is not restricted strictly by eq.(17), small deviation may not cause obvious diversity of the segmentation. To elaborate this problem, we would come back to the image "Bird" shown in Fig. 5(a). Fig. 6 shows how the segmentation result varies with ε. It is known from the result, the larger ε is, the farther the "wave" transmits, the more pixels pulse. That means larger area segmentation, shown in fig.6(a). It should be mentioned that too large may result in over-segmentation, contrarily, too little ε results in undersegmentation, fig.6(d).

So the larger ε is suit for the images with the object intensity in wide range. Ideally, we wish to form three groups of pixels that belong to visually distinct regions such as the branches, the background and the bird.
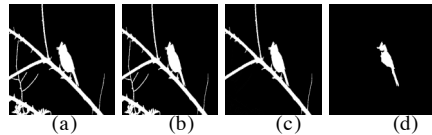

Figure 6 How the segmentation result varies with ε. From left to right $\varepsilon = 20$, $\varepsilon = 15$, $\varepsilon = 10$, $\varepsilon = 3$

5 Experiments and discussion

To evaluate our proposed method, in this section, we, thus, experimentally carry out extensive comparisons with fastlinking model (J.L. Johnson), Threshold Segmentation and Edge Segmentation (Gauss operator) on two types of data. One is the synthetic images, including two synthetic noisy images and three synthetic uneven lighting images; the other is a collection of natural images coming from the Berkeley Segmentation Dataset and some empirically usual tested image database. Both contain a variety of images covering a wide range of viewpoints and illumination angles. In both experiments, only grayscale information between 0 and 255 is used. Appropriate setting of the parameters is a prerequisite for successful segmentation. To make a fair comparison, we tune them over a wide range of values and carefully select "optimum" so that each method presents the perspectively best results among the numerous different partitions for each image. Except for β, the parameters of the NIPCNN and Fastlinking are same, shown in table 2. As analysis in section 2.2, β is an important parameter for fastlinking model, so we select according to each image. But in NIPCNN, it is Fixed value $\beta = 0.5$. The parameters of methods based on threshold, Gaussian are T (threshold), and , respectively.

Table 2. Parameters of NIPCNN and Fastlinking model

α_F	α_L	α_θ	V_F	V_L	V_θ	r
0.001	1	10	0.01	1	2	7

5.1 Experiments on Synthetic Images

In this subsection, to assess the effectiveness of our algorithm for segmentation tasks, we first perform some experiments on synthetic images: the objects with large range of intensity and synthetic noisy images. Figure 7 offers some examples. From up to down, the first row is segmentation for ideal image, with clear contours, even background and smoothly varying intensity; the second row is segmentation for image degraded by uneven lighting; the last two rows are segmentation for Gaussian noisy image (mean 0, standard deviation 35 and 25.5 respectively).
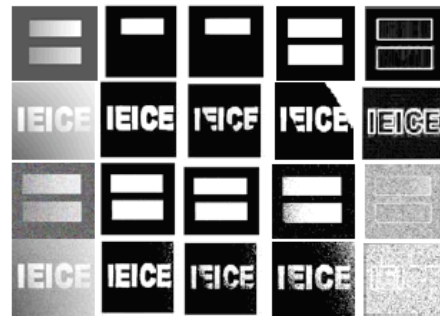
(a)　(b)　(c)　(d)　(e)

Figure 7 Segmentation on synthetic images, from left to right: (a) input images, (b) segmented images by NIPCNN, (c) fastlinking model, from up to down, $\beta = 3$, $\beta = 1$, $\beta = 3$, $\beta = 1$, (d) Threshold, from up to down, T=100; T=212; T=128; T=211 (e) Gaussian.

For images with objects sharply distinguished from background, segmentations based on threshold and edge show validity on the premise of appropriate parameters selection. It is a pity that, these classical methods are slack for noised images or images with uneven background. From the experimental data, the results of NIPCNN are obviously more integrated than classic fastlinking model. In some cases, the integrated segmentation as achieved by NIPCNN may be gotten by the fastlinking in cost of adjusting the parameters. But it is a fussy task to adjust the parameter according to each image. Especially, advantage of NIPCNN on noised images is noticeable. In the mess, NIPCNN has advantage on these synthetic images.

5.2 Experiments on Natural Images

The natural images are more challenging in that they contain significant variations in scale, illumination conditions (sunny versus overcast), material appearance, and sensor noise. The difficult phenomena of shadows, multiple scales, and translucent vegetation are ubiquitous. We wish to segment the natural images into several spatially coherent groups of pixels corresponding to physically objects in the natural world. Here, the representative sample images are divided into two types. One type contains the images composed of the objects include long, thin lines and the background has large homogenous areas, such as the handwriting and branches. The other type comes from the Berkeley Segmentation Dataset which contains a wide range of subject matter, such as the translucent water with surface ripples, the plane in the clouded sky, the birds in the sparse branch,

etc.

In general, our target for this type of images is to correctly extract the objects from the complex uneven lighting backgrounds or equivalently separate the backgrounds from the objects. Fig.8 shows the segmentation results for these images. From left to right, the five columns show the input images [Fig. 8(a)], segmented results by NIPCNN, Fastlinking , Threshold and Gaussian, respectively. It can be seen that NIPCNN separates the complex background [the black area of Fig. 8(b)] well from the objects [the white areas of Fig. 8(b)]. In contrast, Fastlinking, Threshold and Gaussian are disabled by such images. Too small $\hat{a}$ will result in incomplete extraction, while larger $\hat{a}$ will result in unreasonable segmentation. The noise or speckles in the background will raise error rate of Threshold and Gaussian method. This sufficiently indicates that NIPCNN is promising for dealing with these challenging images under the condition of uneven lighting.
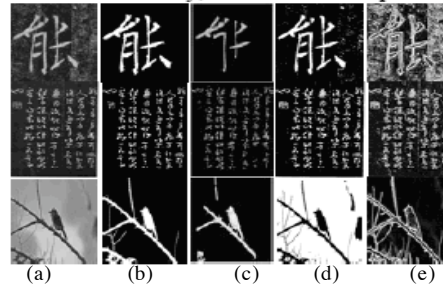

(a)　(b)　(c)　(d)　(e)

Figure 8 Results for several natural images. (a) input images, (b) segmented images by NIPCNN, (c) Fastlinking(J.L. Johnson), $\beta = 8$, $\beta = 10$, $\beta = 3$ (d) Threshold: T=105,T=68,T=114 (e) Gaussian.
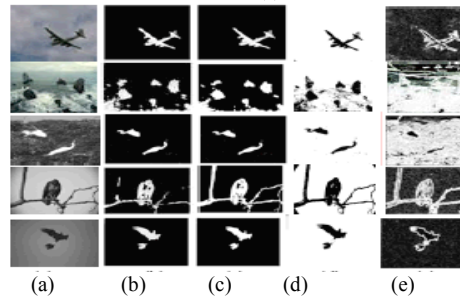

(a)　(b)　(c)　(d)　(e)

Figure 9 Results for several natural images coming from Berkeley database. From left to right, they respectively are (a) input images, (b) segmented images by NIPCNN, (c) Fastlinking(J.L. Johnson), from up to down, $\beta = 1$, (d) Threshold, from up to down, T=49, T=59, T=220, T=108, T=65 (e) Gaussian.

6. CONCLUSION

In this paper, a method for image segmentation by PCNN has been proposed. As in most of the methods using a similar classification mechanism, the derivation of the proposed approach is based on a neighborhood dominant analysis, in which two factors are defined to measure the neighborhood distribution. The combination of two factors is also measure of dissimilarity between the informational contents of segmentation classes, and as a result, between these contents. Then, linking is adaptively modified according to the neighborhood. Cooperating with adjustment threshold within one iterative, the integrated segmentations for different images are achieved with same parameters. Experimental comparison with classical Fastlinking model (J.L. Johnson) and segmentation based on threshold and edge verified the method.

It should be pointed out that, the parameters in this paper may not suit for all images, especially for the images with obvious texture in the objects should be segmented, or the images with inconspicuous contours. For these types images, ε should be adjusted according to each statistical feature of intensity. This is the future work of the present study.

References

[1] R.Ekkhorn, H.J.Reithoeck, hI.Arndt, and P.Dicke, Feature Linking via Synchronization amondg Distributed Assemblies: Simulations of Result from Cat Visual Cortex, Neural computation, vol. 2, pp. 293-307, 1990

[2] John L. Johnson, Pulse-Coupled neural nets: translation, rotation, scale, distortion, and intensity signal invariance for images, App.l.Opt., vol. 33, No.26, 6239-6253, 1994

[3] Johnson J L, Padgett M L. PCNN models and applications. IEEE Trans. on Neural Networks , vol.10, No.3,pp. 480 – 498, 1999

[4] G.Kunimad and H.S.Ranganath, Perfect Segmentation Using Pulse Coupled Neural Networks, IEEE, vol.10, No.3, pp. 591-598, 1999

[5] G. Kuntimad and H. S. Ranganath. Perfect Image Segmentation Using Pulse Coupled Neural Networks. IEEE Trans. on Neural Networks, vol.10, vol.3, pp.591-598, 1999

[6] Zhang Junying, Lu Zhijun, Shi Lin, Dong Jiyang. Filtering images contaminated with pep and salt type noise with pulse_coupled neural networks. Science in China Ser. F Information Sciences, vol. 48, No.3, pp.322-334, 2005

[7] Daisuke Yamaoka, Yuuki Ogawa, Kosei Ishimura, Motion Segmentation Using Pulse-Coupled Neural Network. SICE kmual Conference in Fukui, August, 2003, pp. 2778-2783

[8] Broussard, R. P., Rogers, S. K., Oxley, M. E. et al., Physiologically motivated image fusion for object detection using a pulse coupled neural network, IEEE Trans. Neural Networks, vol. 10, No.3 pp. 554-563, 1999

[9] Caufield, H. J., Kinser, J. M., Finding the shortest path in the shortest time using PCNN's, IEEE Trans. Neural Networks, vol.10, No.3, pp. 604-606, 1999

[10] Bi Yingwei, Qiu Tianshuang. An Adaptive Image Segmentation Method Based on a Simplified PCNN. Acta Electronica Sinica., vol. 33, No.4, pp. 647-650, 2005

[11] Geza Szekely, Thomas Lindblad, E. Stella, A. Distante, Parameter adaptation in a simplified Pulse-Coupled Neural Network, Ninth Workshop on Virtual Intelligence/Dynamic Neural Networks, ,vol. 3728, pp.278-285, 1999

[12] Ma Yide, Dai Ruolan, Li Lian. Automated image segmentation using pulse coupled neural networks and image's entropy. Journal on Communications, vol. 23, No.1, pp. 46-51, 2002

[13] Wei Huang a, Zhongliang Jing. Multi-focus image fusion using pulse coupled neural network. Pattern Recognition Letters, vol. 28 pp.1123-1132, 2007

[14] A. Falcão, J. Stolfi, and R. Lotufo, The image foresting transform: Theory, algorithms, and applications. IEEE Trans. Pattern Anal. Mach. Intell., vol. 26, no. 1, pp. 19–29, Jan. 2004.

[15] Jundi Ding, Runing Ma, and Songcan Chen, A Scale-Based Connected Coherence Tree Algorithm for Image Segmentation. IEEE Trans. Image Processing, vol. 17, No. 2, 2008.