Using Neuro-Fuzzy Networks with Immune Particle Swarm Optimization to Compensate Backlight Image

Cheng-Jian Lin *, *Member, IEEE*, and Yong-Cheng Liu

***Abstract*—The digital camera is very popular and general product in recent year. Some problems might happen, when we capturing digital photos by CCD, such as backlight, over-saturation, low contrast. Image compensation plays an important role in the amendment of images with some flaws. However, color image compensation still not achieves satisfactory performance. We proposed a new technique to compensate the backlight images. There are two processing stages, the detection stage and the compensation stages in the technique. In detection stage, we transferred the color space to gray space by feature weighting, then used the proposed Functional-Link Neuro-Fuzzy Network (FNFN) with Immune Particle Swarm Optimization(IPSO) for detecting compensation degree by two backlight factors. In compensation stage, we also proposed the adaptive cubic curve method to compensate and enhance the brightness of backlight images according to the compensation degree of each image, and the backlight degree indicated by histograms of the luminance distribution in detection stage. The experiment result was conducted to show the backlight images can be compensated effectively.**

Keywords: backlight compensation, neuro-fuzzy networks, immune algorithm, particle swarm optimization.

I. INTRODUCTION

Lately, Digital Cameras (DC) are getting universal in our day to day life. They even become one of the most essential functions in mobile phones. The techniques of the digital cameras have also been improved, for example, the against the hand to share, the light compensation, to focus a object...etc. However, there are some problems with the image retrieving which will still need to be improved. Our paper is focus on addressing an effective compensation for the backlight problem which is caused by the main accessories between the light and the lens, when the photos are taken.

Recently, some research about backlight image compensation have be reported, but the numbers are very scanty [1-2]. Lin and Huang [1] proposed a two-stage compensation technique for improving the appearance of the pictures. They utilize the fuzzy c-means learning mechanism and the fuzzy logic rule inference to compensate the back-light images. For attacking the weaknesses of conventional backlight image processing methods, such as over-saturation and diminished contrast, Lin and Chin [2] proposed a backlight image detection and compensation algorithm with fuzzy logic and adaptive compensation curve. The two methods of the above-mentioned are also use fuzzy logic and two backlight factors to determine the compensation value of backlight images, but they employed difference way to extracted the two backlight factors from backlight images.

In this article, there are two major stages in our method, which are the backlight level detecting and backlight image compensation. First of all, in the backlight prediction step, we will transfer images to gray value, and then use pixel for the clustering which will separate the backlight object and the background. We can gather two backlight factors from the information which is in result of the clustering and the gray histogram. We then apply these two backlight factors to the prepared neuro-fuzzy network, which is named the functional-link-based neural fuzzy network (FNFN) [3] to conclude the image compensation value. We will use the immune particle swarm optimization algorithm (IPSO) [4] to training FNFN model in advance. Secondly in the backlight compensation stage, based on the deductive compensation value of the FNFN, we can gather a compensation curve from the backlight image. After the backlight image is transferred through the curve function, it will be acquired a compensation image. Then following by applying this image with this technique and obtain a backlight compensation, we will not have to worry about the backlight problem which is caused by the main accessories between the light and the lens, when the photos are taken. This is because all backlight images can be compensated.

In our simulations, we have actually applied our above discussed techniques to several backlight images. The proposed method performs the accurate detection of the backlight degree and strong compensation effect in backlight images.

II. THE PROPOSED COMPENSATION METHOD

In this section, we will introduce the proposed approach for backlight image compensation. The flow chart of the proposed algorithm is shown in Fig. 1.

* C. J. Lin is with the Dept. of Electrical Engineering National University of Kaohsiung, Kaohsiung, 811 Taiwan, R.O.C. (corresponding author to provide phone: 04-23323000-3131; e-mail: cjlin@mail.cyut.edu.tw).

Y. C. Liu is with the Dept. of Computer Science and Information Engineering Chaoyang University of Technology, Taichung, 413 Taiwan, R.O.C. (e-mail: s9527603@mail.cyut.edu.tw).

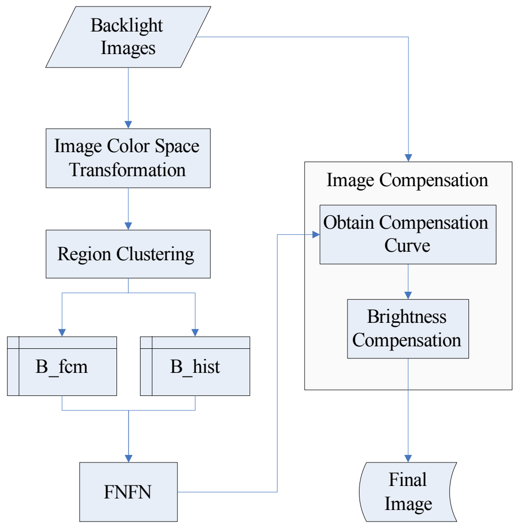

Fig. 1. Flow chart of the proposed backlight compensation method.

A. *Image color space transformation*

In order to satisfy the follow-up two unit: the image histogram and the region clustering, we change images color space from the RGB color model to the YIQ model, and we adopt to operate on the luminance Y component only. This design choice takes advantage of the fact that the human eyes are less sensitive to quantize errors affecting the chrominance components of the image. Eq. (1) shows the color space transformation from RGB to YIQ.

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} R \\ G \\ B \end{bmatrix} \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.275 & -0.321 \\ 0.212 & -0.523 & 0.311 \end{bmatrix} \tag{1}$$

B. *Determine two factors of the backlight image*

*Backlight factor B*_hist

In a backlight image, the contrast between background and backlight objects is usually very great. Through observation of the backlight image histogram, we can find a phenomenon, the distribution of the luminance between background and backlight object that presents the obvious distance. Figure 2 shows a backlight image and its histogram.

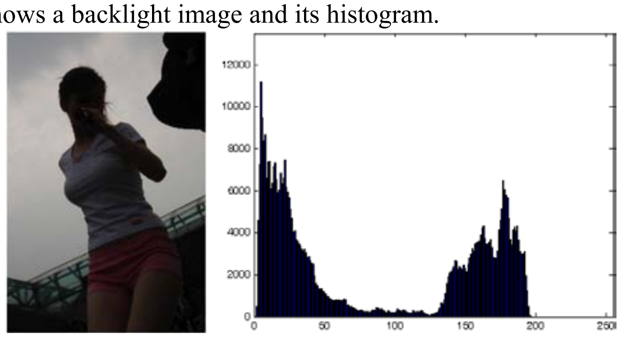

Fig.2.The example of the backlight image and histogram.

Therefore, in order to measure the gradient between background and backlight object in HIST histogram, we utilize sliding window (SW) to deal with the *B_hist* factor. HIST is define as the ratio between the number of pixels whose brightness is higher than a threshold value and the total number of pixels in the whole image. First, we apply the SW to calculate the max SW, when the accumulation of the HIST is smaller than 0.2. Figure 3 show the HIST histogram and the max SW:

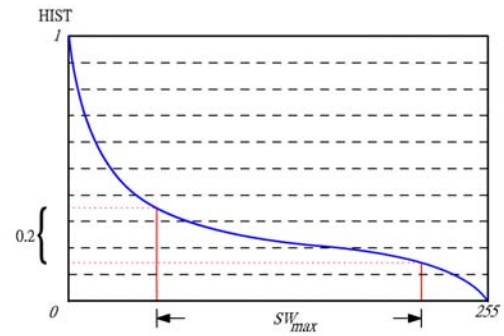

Fig. 3. The HIST histogram and the max window.

We can find a phenomenon through observing, the backlight degree will raise with the SW_{max}. So, we define the *B_hist* as following equation:

$$B\_hist = T_{hist}\left(\frac{SW_{max}}{255}\right) \tag{2}$$

where $T_{hist}(.)$ was a transfer function which transformed *B_hist* into a fuzzy degree. The transfer function is show in the list:

$$T_{hist}(x) = \begin{cases} (x-0.3)/(0.6-0.3) & ,if \quad 0.6 > x > 0.3 \\ 1 & ,if \quad x >= 0.6 \\ 0 & ,otherwise \end{cases} \tag{3}$$

Backlight factor B_fcm

Another backlight factor, we also use the cluster result extract by fuzzy c-means (FCM) [5] to determine it. Separately, the background and backlight object represent the bright and dark area separately in backlight images. Therefore, for the sake of segmenting background and backlight object, the cluster number is set to 2. After FCM algorithm, we will obtain two clustering centroid C1and C2. According to the image histogram information, the luminance can't be accumulated in histogram area between the background and the backlight object, In other word, the smaller accumulating amount of luminance between the background and backlight object is, then the higher the backlight degree is. Based on the characteristic of the above, we can define another backlight factor, *B_fcm* for determining the backlight degree of an image:

$$\text{B\_fcm} = T_{fcm}\left((C_2 - C_1) * \frac{p(C_1) + p(C_2)}{2} - \sum_{i=C_1}^{C_2} p(r_i) \right) \tag{4}$$

where $p(r_i) = \frac{n_i}{n}$ is the probability of the *i*-th gray level, when n was the total number of pixels in the image and n_i was the number of times the level appeared in the image, $T_{fcm}(.)$ was a transfer function which transformed B_*fcm* into a fuzzy degree. The transfer function is show in the list:

$$T_{fcm}(x) = \begin{cases} (0.8 - x)/(0.8 - 0.25) & ,if \quad 0.8 > x > 0.25 \\ 0 & ,if \quad x >= 0.8 \\ 1 & ,otherwise \end{cases} \tag{5}$$

The factor, *B_fcm*, expresses the accumulation of the luminance between two cluster centroid *C1*and *C2*, and the backlight degree is growing with *B_fcm*.

C. Functional-link based neural fuzzy network

This subsection describes the FNFN model, which uses a nonlinear combination of input variables. Each fuzzy rule corresponds to a sub-FLNN, comprising a functional link. Figure 4 presents the structure of the proposed FNFN model.

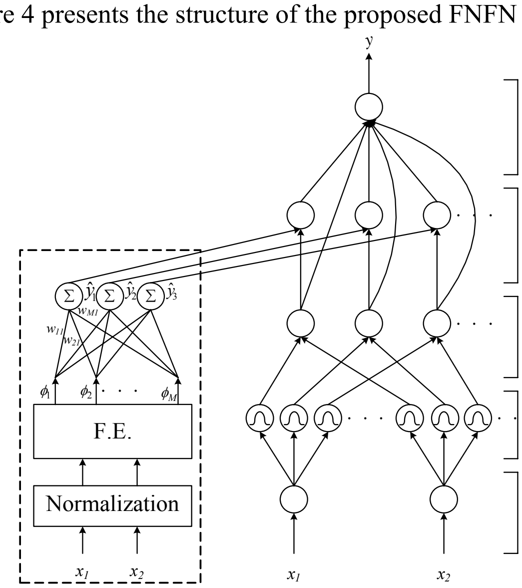

Fig. 4. Structure of proposed FNFN model.

The FNFN model realizes a fuzzy if-then rule in the following form:

Rule-j:

IF x_1 is A_{1j} and x_2 is A_{2j} ... and x_i is A_{ij} ... and x_N is A_{Nj}

$$\text{THEN } \hat{y}_j = \sum_{k=1}^{M} w_{kj}\phi_k = w_{1j}\phi_1 + w_{2j}\phi_2 + ... + w_{Mj}\phi_M \tag{6}$$

where x_i and $\hat{y}_j$ are the input and local output variables, respectively; A_{ij} is the linguistic term of the precondition part with Gaussian membership function; N is the number of input variables; w_{kj} is the link weight of the local output; ϕ_k is the basis trigonometric function of the input variables; M is the number of basis function, and *Rule-j* is the *j*-th fuzzy rule.

The operation functions of the nodes in each layer of the FNFN model are now described. In the following description, u(l) denotes the output of a node in the *l*-th layer.

Layer 1 (Input Node): No computation is performed in this layer. Each node in this layer is an input node, which corresponds to one input variable, and only transmits input values to the next layer directly:

$$u_i^{(1)} = x_i \tag{7}$$

Layer 2 (Membership Function Node): Nodes in this layer correspond to a single linguistic label of the input variables in Layer 1. Therefore, the calculated membership value specifies the degree to which an input value belongs to a fuzzy set in layer 2. The implemented Gaussian membership function in layer 2 is

$$u_{ij}^{(2)} = \exp\left(-\frac{[u_i^{(1)} - m_{ij}]^2}{\sigma_{ij}^2} \right) \tag{8}$$

where m_{ij} and σ_{ij} are the mean and variance of the Gaussian membership function, respectively, of the *j*-th term of the *i*-th input variable x_i.

Layer 3 (Rule Node): Nodes in this layer represent the preconditioned part of a fuzzy logic rule. They receive one-dimensional membership degrees of the associated rule from the nodes of a set in layer 2. Here, the product operator described above is adopted to perform the IF-condition matching of the fuzzy rules. As a result, the output function of each inference node is

$$u_j^{(3)} = \prod_i u_{ij}^{(2)} \tag{9}$$

where the $\prod_i u_{ij}^{(2)}$ of a rule node represents the firing strength of its corresponding rule.

Layer 4 (Consequent Node): Nodes in this layer are called consequent nodes. The input to a node in layer 4 is the output from layer 3, and the other inputs are nonlinear combinations of input variables from a functional link neural network, where the nonlinear combination function has not used the function tanh(.), as shown in Fig. 4 For such a node,

$$u_j^{(4)} = u_j^{(3)} \cdot \sum_{k=1}^{M} w_{kj}\phi_k \tag{10}$$

where w_{kj} is the corresponding link weight of functional link neural network and ϕ_k is the functional expansion of input variables. The functional expansion uses a trigonometric polynomial basis function, given by $\lfloor x_1 \sin(\pi x_1) \cos(\pi x_1) x_2 \sin(\pi x_2) \cos(\pi x_2) \rfloor$ for two-dimensional input variables. Therefore, M is the number of basis functions, $M = 3 \times N$, where N is the number of input variables.

Layer 5 (Output Node): Each node in this layer corresponds to a single output variable. The node integrates all of the actions recommended by layers 3 and 4 and acts as a defuzzifier with,

$$y = u^{(5)} = \frac{\sum_{j=1}^{R} u_j^{(4)}}{\sum_{j=1}^{R} u_j^{(3)}} = \frac{\sum_{j=1}^{R} u_j^{(3)} \left(\sum_{k=1}^{M} w_{kj}\phi_k \right)}{\sum_{j=1}^{R} u_j^{(3)}} = \frac{\sum_{j=1}^{R} u_j^{(3)} \hat{y}_j}{\sum_{j=1}^{R} u_j^{(3)}} \tag{11}$$

where R is the number of fuzzy rules, and y is the output of the FNFN model.

As described above, the number of tuning parameters for the FNFN model is known to be $5 \times N \times R$, where N and R denote the number of inputs and existing rules, respectively.

D. *Immune particle swarm optimization algorithm*

This subsection describes an efficient immune particle swarm optimization (IPSO) [4] learning method for the FNFN model design. Analogous to the biological immune system, the proposed algorithm has the capability of seeking feasible solutions while maintaining diversity. The proposed IPSO is combining the immune algorithm (IA) and the particle swarm optimization (PSO) to perform parameter learning. In order to avoid trapping in a local optimal solution and to ensure the search capability of a near global optimal solution, mutation plays an important role in IPSO. Therefore we employed the advantages of PSO to improve the mutation mechanism of immune algorithm. The whole learning process is described step-by-step below.

Coding Step

The coding scheme consists of the coding done by the IPSO. The IPSO codes the adjustable parameters of a FNFN into an antibody, as shown in Fig. 5, where MS_i represents the parameters of the antecedent of the i-th rule in the FNFN and C_i represents the parameters of the consequent of the i-th rule, respectively. In this paper, a Gaussian membership function is used with variables representing the mean and deviation of the membership function. Each fuzzy rule in Fig. 5 has the form in Eq. (6), where m_{ij} and σ_{ij} represent a Gaussian membership function with mean and deviation of the j-th dimension and i-th rule node and w_{im} represents the corresponding parameters of consequent part, and m is equal the M in Eq. (6).

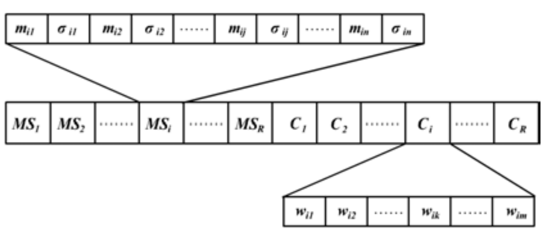

Fig. 5. Coding a FNFN into an antibody in the IPSO method.

Initial Population Production

In the immune system, the antibodies are produced in order to cope with the antigens. In other words, the antigens are recognized by a few of high affinity antibodies. All of the initial antibodies utilizing a real variable string are generated by random.

Calculate Affinity Values

For the large number of various antigens, the immune system has to recognize them for their posterior influence. In biological immune system, affinity refers to the binding strength between a single antigenic determinants and an individual antibody-combining site. The process of recognizing antigens is to search for antibodies with the maximum affinity with antigens. In this paper, the affinity value is designed according to the follow formulation:

$$\textit{Affinity value} = \frac{1}{\sqrt{\frac{1}{N_t}\sum_{k=1}^{N_t}\left(y_k - y_k^d\right)^2}} \tag{12}$$

where y_k represents the k-th model output, y_k^d represents the desired output, and N_t represents the number of the training data. In the problems, the higher affinity refers to the better antibody.

Production of Sub-antibodies

In this step, we will generate several neighborhoods to maintain solution variation. This strategy can prevent the search process from becoming premature. We can generate several clones for each antibody on feasible space by Eq. (13) and Eq. (14). Each antibody regards as parent while the clones regard as children (sub-antibodies). In other words, children can regard as several neighborhoods of near parent.

mean and deviation :

$$clons\,[children_{i\_c}] = antibody\,[parent_i] + \alpha \tag{13}$$

weight :

$$clons\,[children_{i\_c}] = antibody\,[parent_i] + \beta \tag{14}$$

where $parent_i$ represents the i-th antibody from the antibody population; $children_{i\_c}$ represents clones number c from the i-th antibody; α and β are parameters that control the distance between parent.

Mutation of Sub-antibodies Based on Particle Swarm Optimization

In order to avoid trapping in a local optimal solution and to ensure the search capability of near global optimal solution, mutation plays an important role in IPSO. Through the mutation step, only one best child can survive to replace its parent and enter the next generation. Hence, we employed the advantages of particle swarm optimization (PSO) to improve mutation mechanism.

PSO is not only a recently invented high-performance optimizer that is very easy to understand and implement, but it also requires little computational bookkeeping and, generally, only a few lines of code. Each particle has a velocity factor $\vec{v}_i$ and a position factor $\vec{x}_i$ to represent a possible solution. The velocity for each particle is updated by

$$\begin{aligned}\vec{v}_i(k+1) &= \omega * \vec{v}_i(k) \\ &\quad + \phi_1 * rand() * (Lbest - \vec{x}_i(k)) \\ &\quad + \phi_2 * rand() * (Gbest - \vec{x}_i(k))\end{aligned} \tag{15}$$

where ω is the coefficient of inertia, ϕ_1 is the cognitive study, and ϕ_1 is the group study. The *rand()* is uniformly distributed random numbers in [0, 1]. The term $\vec{v}_i$ is limited to the range

$\pm\vec{v}_{max}$. If the velocity violates this limit, it will be set at its proper limit. Changing velocity enables every particle to search around its individual best position and global best position. Based on the updated velocities, each particle changes its position according to the following:

$$\vec{x}_i(k+1) = \vec{x}_i(k) + \vec{v}_i(k+1) \tag{16}$$

When every particle is updated, the affinity value of each particle is calculated again. If the affinity value of the new particle is higher than those of local best, then the local best will be replaced with the new particle.

Promotion and Suppression of Antibodies

In order to affect antigens and keep diversity to a certain degree, we use information entropy theory to measure the diversity of antibodies. If the affinity between two antibodies is greater than the suppression threshold Th_{aff}, these two antibodies are similar, and the antibody of lower affinity value is reduced a small amount of value λ. The antibodies with high antigenic affinity are transformed into long-lived B memory cells; others with low antigenic affinity are affected. Fig. 6 shows the immune algorithm composed of N antibodies having L genes.

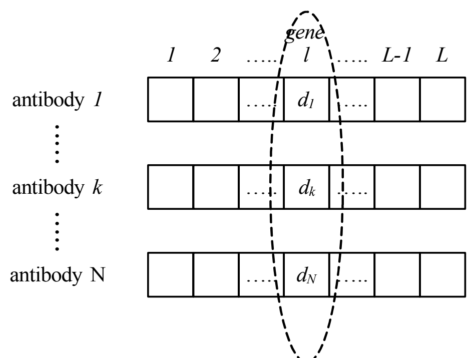

Fig. 6 The coding of antibody population.

From information entropy theory, we get

$$IE_l(N) = \sum_{i=1}^{N} - P_{il} \log P_{il} \tag{17}$$

where P_{il} is the probability that the i-th allele comes out at the l-th gene. The diversity of the genes is calculated using Eq. (17). The average entropy value *IE(N)* of diversity can be also computed as follows:

$$IE(N) = \frac{1}{L} \sum_{l=1}^{L} IE_l(N) \tag{18}$$

where L is the size of the gene in a antibody. There are two kinds of affinities in IPSO. One explains the relationship between an antibody and an antigen using Eq. (12). The other accounts for the degree of association between the j-th antibody and the k-th antibody and measures how similar these two antibodies are. It can calculated by using

$$Affinity\_Ab_{jk} = \frac{1}{1+IE(2)} \tag{19}$$

Elitism Selection

When a new generation is created, the risk of losing the best individuals is always existent. In this study, we adopt elitism selection to overcome the above-mentioned problem. Therefore, the antibodies are ranked in ascending order to their affinity value. The best individuals are kept as the parent for the next generation. Elitism selection improves the efficient of IPSO considerably, as it prevents losing the best result.

E. Image compensation

In the previous sub-section, we proposed a procedure to detect the backlight degree of an image. After the backlight degree is detected, we can compensate the backlight image according to the detected backlight degree. In light of the characteristic of the contrast degree in backlight image, we have to increase the luminance of backlight object and reduce the luminance of background. Therefore, this paper depends on the adaptive cubic curve method which is determined by the backlight degree to compensation a backlight image. The element that the curve constitutes includes the upward and downward parabolic curves, and the curve can be adjustable by tuning point (*TP*). The curve shown in the following figure:

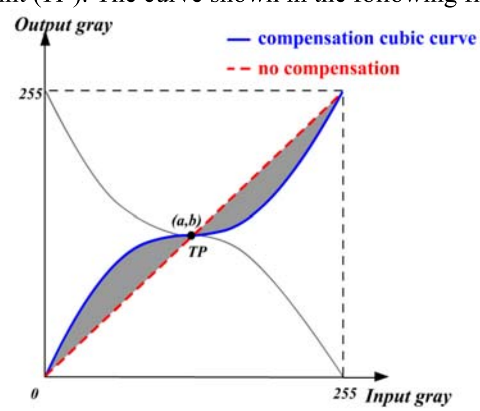

Fig. 7. An adaptive cubic curve.

The tuning point (a,b) is determine by backlight degree which is computed through a well-trained FNFN. The cubic curve which is described by Eq. (20):

$$f(x) = \begin{cases} \frac{-b}{a^2}(x-a)^2 + b & , if\ x < a \\ \frac{(255-b)}{(255-a)^2}(x-a)^2 + b & ,otherwise \end{cases} \tag{20}$$

The above-mentioned equation is combined with the half-upward curve and the half-downward parabolic curves, and the value a and b is calculated as:

$$\begin{cases} a = (C_1 + C_2)/2 \\ b = a + (B_{degree} \times (C_2 - a)) \end{cases} \tag{21}$$

where C_1 and C_2 are the cluster center which are obtained by FCM algorithm; B_{degree} is the backlight degree which is obtained by FNFN. Finally, we can get a compensated image, when all pixels in backlight image are transformed via the adaptive cubic curve.

III. RESULTS

This paper applied the FNFN with IPSO to estimate the backlight degree of a backlight image, and the arguments of FNFN and IPOS are shown in table 1. In this section we compensated three backlight images for testing the efficiency of the proposed method. Fig. 8(a), Fig. 9(a) and Fig. 10(a) are the original backlight images. Fig. 8(b), Fig. 9(b) and Fig. 10(b) are the compensated images.

Table 1:The initial parameters before training

Parameters	**Value**
Antibody Population Size	50
Coding Type	Real Number
Clones number *c*	5
Crossover rate	0.8
Mutation rate	0.3
ω	0.25
ϕ_1	0.8
ϕ_2	1.25
Suppression threshold (Th_{aff})	0.8

(a) (b)

Fig. 8(a) original image(b) compensated image.

(a) (b)

Fig.9(a) original image(b) compensated image.

(a) (b)

Fig. 10(a) original image(b) compensated image.

IV. CONCLUSION

In this paper, a backlight image compensation method using FNFN with IPSO and the adaptive multi-cubic curve is proposed. In the backlight level detecting stage, we have extracted two operative backlight factors, and the FNFN model is based on the two factors that can accurately detect the compensation degree. In the compensation stage, we using adaptive compensation curve effectively to improve image backlight problem and to accord with the generalization state of backlight images.

ACKNOWLEDGEMENT

This research was sponsored by National Science Council, R.O.C., under grant NSC95-2221-E-324-028-MY2.

REFERENCES

[1] D. T. Lin and C. C. Huang, "Image Back-light Compensation with Fuzzy C-Means Learning Algorithm and Fuzzy Inferencing," *ISSPA2003*, Paris, France, Jul, 2003.

[2] C. L. Chin and C. T. Lin, "Detection and Compensation Algorithm for Backlight Images with Fuzzy Logic and Adaptive Compensation Curve," *International Journal of Pattern Recognition and Artificial Intelligence,* vol. 19, no. 8, Mar, 2005, pp. 1041-1057.

[3] C.H. Chen, C.T. Lin, and C.J. Lin, "A Functional- Link-Based Fuzzy Neural Network for Temperature Control," *2007 IEEE Symposium on Foundations of Computational Intelligence(FOCI 2007)*, Honolulu, Hawaii, USA, Apr, 1-5, 2007, pp. 53-58.

[4] C. Y. Lee, C. J. Lin, and H. C. Chuang, "Efficient Immune-Based Particle Swarm Optimization Learning for TSK-Type Neuro-Fuzzy Network Design," *The 17th Workshop on Object-Oriented Technology and Applications,* Chang Gung University, Tao-Yuan, Taiwan, Sep, 12, 2006.

[5] N. R. Pal, J. C. Bezdek, "On Cluster Validity for the Fuzzy C-means Model," *IEEE TRANSACTIONS ON FUZZY SYSTEMS.* vol. 3, no. 3, Aug, 1995.