

A Neuro-Fuzzy Approach for Compensating Color Backlight Images

MU-CHUN SU*, YUAN-SHAO YANG, JONATHAN LEE and
GWO-DONG CHEN

*Department of Computer Science & Information Engineering, National Central University,
Taiwan, R.O.C. e-mail: muchun@csie.ncu.edu.tw*

Abstract. This paper presents a neuro-fuzzy approach for compensating exposure in the case of backlighting, regardless of the position of objects. To achieve the compensation effect, the fuzzy C-means algorithm is first used to extract features from a backlight image. Then these extracted features are presented to a trained artificial immune system based neuro-fuzzy system (AISNFS) to estimate the amount of compensation. Finally, the estimated amount of compensation incorporated with a compensation equation is used to enhance the intensity component of the backlight image to produce a compensated image. Several backlight images were used to test the performance of the algorithm.

Key words. backlight image, fuzzy clustering algorithm, histogram equalization, image enhancement, neuro-fuzzy system

1. Introduction

Although most of the digital cameras have many appealing features, such as auto focus, auto exposure, etc, which enable users to easily take good pictures under various shot conditions, users still have chances of taking backlight images. When the luminance difference between the main object and the background is high, backlight images, such as Figure 1(a), are usually produced.

There are many different approaches for avoiding taking backlight pictures. Some approaches involve in the design of an auto exposure (AE) system to optimize the image luminance by controlling the exposure. Due to the introduction of digital circuits into digital cameras, it has been possible for an AE system to detect whether the backlighting situation is going to happen and compensate the exposure to improve the luminance of the main object. Currently, there are several different exposure control systems, and most of them are based on fuzzy rules. For example, Morimura proposed an exposure control scheme based on the distribution of luminance for multiple regions of a screen [1]. Haruki and Kikiuchi proposed a method of dividing a screen into six regions and weighting the luminance data of each region to put emphasis on the center of the screen [2]. Murakami and Honda proposed an exposure control system which uses ‘hue’ and ‘chroma’ of pixels to derive the importance of the background, and determines a compensation

*Corresponding author.

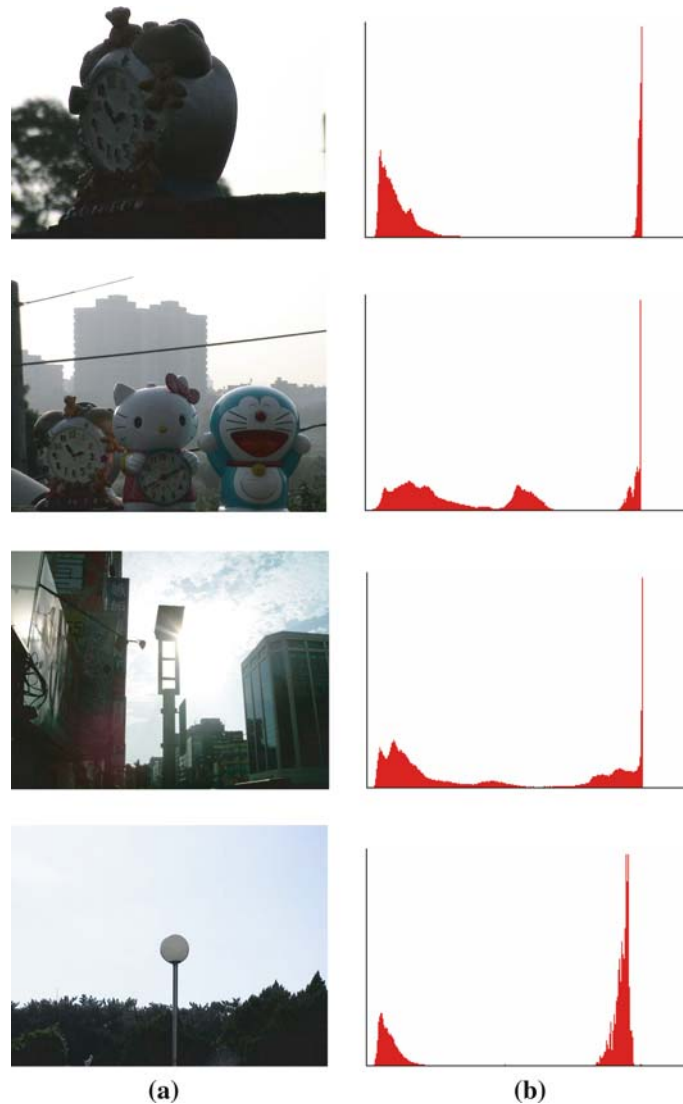


Figure 1. Backlight images. (a) Original backlight images, (b) their histograms.

amount by fuzzy reasoning [3]. Shimizu *et al.* proposed an algorithm based on 25 fuzzy rules for compensating exposure, regardless of the position of objects [4]. Each has its own merits and disadvantages.

In addition to the afore-mentioned approaches, some approaches focus on post-processing. Contrast enhancement techniques can be used to improve backlight images. There are many contrast enhancement algorithms [5–11], and among these, histogram equalization [5–9] is the most popular automatic method due to its simplicity and effectiveness. Global histogram equalization uses the histogram information of an image as its transformation function that stretches the contrast

of the high histogram region and compresses the contrast of the low histogram region. This is less effective where the contrast characteristics vary across the image. To overcome this limitation, some adaptive histogram equalization algorithms were proposed [12–17]. Basically, these algorithms generate the mapping for each pixel from the histogram in a surrounding window. The price paid for the improvements is the high computation complexity. In addition, some of these algorithms require very complex computation and hardware, so they cannot be applied to real-time commercial products.

In this paper, we propose a new algorithm for compensating backlight images. The details of the proposed algorithm will be discussed in Section 2. An introduction to an artificial immune system based neuro-fuzzy system (AISNFS) will be given in next section. In Section 4 we will present the simulation results. Finally Section 5 concludes this paper.

2. The Algorithm for Compensating Backlight Images

In our previous work [18], we have proposed a neural-network-based compensation algorithm. In this paper we propose a more effective approach. For backlight images, the brightness of the background is significantly higher than that of the object. That is, backlight images have a tendency that the brightness histograms of these images usually separate into a bright region and a dark region, as shown in Figure 1(b). A simple method to compensate a backlight image is to directly add the same amount of compensation to each pixel. For this simple method, the only problem to be solved is how to decide an appropriate amount of compensation for each backlight image. The price paid for the simplicity is that the qualities of compensated images are image-dependent, as shown in Figure 2(b). This simple method usually distorts the color information of the backgrounds in the images. For example, the blue sky in the fourth image shown in Figure 2(b) turns into a white sky. In addition, the contrast of the main objects is not enhanced. To avoid the loss of the color information in backgrounds, we may just add a certain amount of compensation to each pixel belonging to the main objects. Although this idea seems correct the compensated images (as shown in Figure 2(c)) sometimes become even worse. For example, the color of the area surrounding those dolls in the second image shown in Figure 2(c) turns into gray. The reason is due to the fact that it is difficult to segment the main objects from the images.

Therefore a crucial problem is to determine how much the compensation value should be added to the dark region so as to simultaneously make the main object look brighter, keep the color information of the background from being distorted, and enhance the contrast of the compensated image. Here, we propose a simple approach to compensating backlight images. The block diagram of the proposed algorithm for compensating backlight images is shown in Figure 3. The algorithm involves the following four steps:

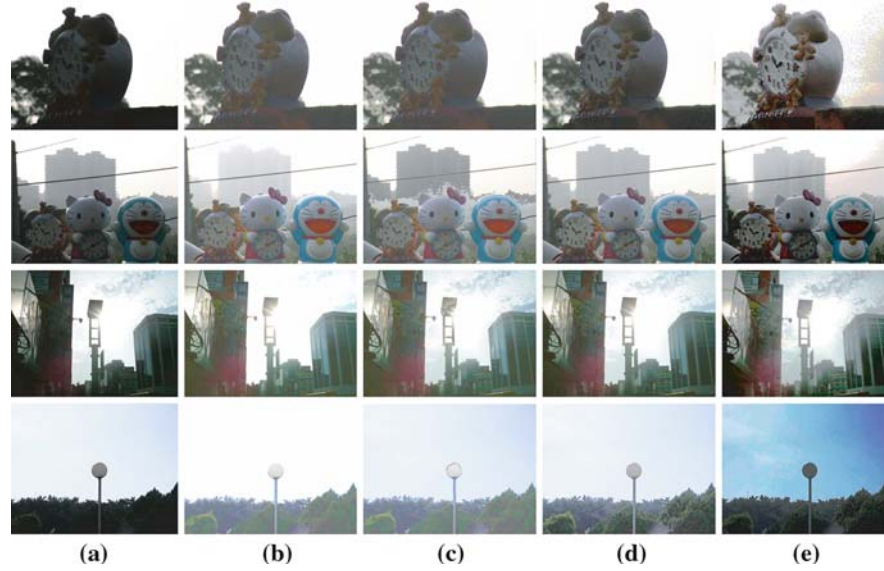


Figure 2. Compensation results. (a) The left column consists of the original image, (b) the second column consists of the images compensated by adding the same compensation amount into each pixel, (c) the central column consists of the images compensated by adding the same compensation amount into each pixel belonging to the main objects, (d) the fourth column consists of the images compensated by the proposed algorithm, (e) the right column consists of the histogram-equalized images.

STEP 1: COLOR MODEL TRANSFORMATION

Three popular color models for image processing are the RGB, the YIQ, and the HIS models. Each model has its own considerations and merits. The HIS model has two important characteristics: (1) the intensity component, I , is decoupled from the color information in the image and (2) the hue and saturation components are intimately related to the way in which human being perceive color. While the hue component, H , is a color attribute that describes a pure color, the saturation component, S , gives a measure of the degree to which a pure color is diluted by white light. Since the I component can provide information about the brightness and it is decoupled from the color information in the image these two features make the HIS model an ideal tool for us. Therefore, the first step in our algorithm is to transform the color image into the HIS model.

STEP 2: FEATURE EXTRACTION

Feature extraction plays a very important role in the compensating algorithm. Understandably, computational complexity will increase with the number of features used to describe a backlight image.

Among the three components of the HIS color model only the I component deals with the brightness; therefore, the following feature extracting procedure will be based on the I component only. Here, we propose to use the means and the

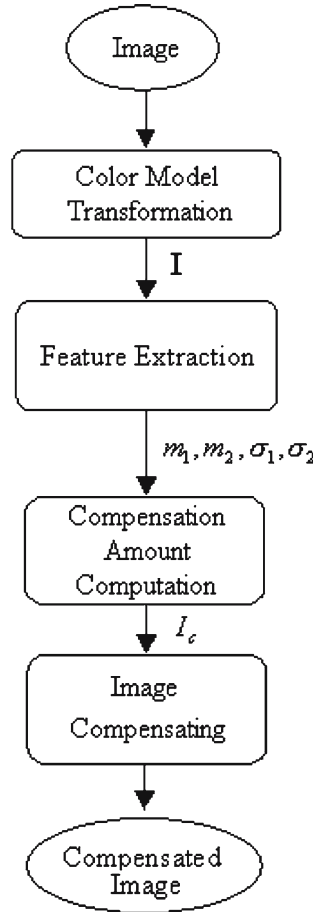


Figure 3. The block diagram of the proposed algorithm for compensating backlight images.

variances of the dark and the light regions in the brightness histogram as the features. First, we use a clustering algorithm such as the FCM algorithm [19, 20] to partition the brightness histogram into two clusters corresponding to the dark region and the bright region, respectively. The mean, m_i , and the variance, σ_i , corresponding to each region are computed as follows:

$$m_i = \frac{\sum_{j=1}^{M \times N} I(j) u_{ij}}{\sum_{j=1}^{M \times N} u_{ij}} \quad \text{for } i = 1, 2 \quad (1)$$

and

$$\sigma_i^2 = \frac{\sum_{j=1}^{M \times N} (I(j) - m_i)^2 u_{ij}}{\sum_{j=1}^{M \times N} u_{ij}} \quad \text{for } i = 1, 2 \quad (2)$$

where $M \times N$ is the image size, $I(j)$ is the I component of pixel j , and u_{ij} represents the degree of pixel j belonging to the i th cluster. The first and the sec-

ond clusters correspond to the dark region (the main objects) and bright region (the background), respectively. These four features, the means, m_1 and m_2 , and the variances, σ_1 and σ_2 , are then presented to a neuro-fuzzy system to estimate the appropriate amount of compensation for the backlight image.

STEP 3: COMPENSATION AMOUNT COMPUTATION

In [4], 25 manually selected fuzzy rules were used to compute the amount of compensation. In this paper, we choose to collect a training set to train a function approximator to compute the appropriate amount of compensation for each backlight image. Many different neural networks, such as multi-layer perceptrons (MLPs), radial basis function (RBF) networks, etc, can be used as function approximators here. In addition to these two popular neural networks, we propose to use an artificial immune system based neuro-fuzzy system (AISNFS) to estimate the amount of compensation. An AISNFS is an on-line learning neuro-fuzzy system which was inspired by part of the mechanisms in immune systems. A detailed introduction to AISNFS will be given in next section.

STEP 4: IMAGE COMPENSATING

If the amount of compensation resulting from the previous step is directly added to the intensity component of each pixel the quality of the compensated image may not look good enough because the intensity range of the main objects is still not wide enough. Instead of directly adding the same amount of compensation to each pixel, we add different amount of compensation to each pixel based on its brightness. Our idea is as follows. In a backlight image, the brightness histogram has a tendency of exhibiting a dark cluster and a bright cluster. We use the degree of a pixel belonging to the bright cluster (i.e. background), u_{2j} , to decide the relative amount of compensation. Besides the addition of the compensation amount, a contrast enhancement procedure is carried out simultaneously.

Suppose the amount of compensation computed by a trained AISNFS is I_c . Let the smallest I component in the original backlight image be denoted as $I_{\min}$ and its corresponding degree of belonging to the bright cluster be denoted as $u_{2\min}$. In addition, the value of u_{2m_1} represents the degree of the mean value of the dark cluster, m_1 , belonging to the bright cluster. The backlight image is compensated through the following two stages:

(1) Stage One: Adding Compensation Amount

$$I_j = I_j + I_c \times (1 - u_{2j}) \quad \text{for } 1 \leq j \leq M \times N \quad (3)$$

Obviously, the smaller the value of u_{2j} is the larger the amount of the compensation is.

(2) Stage Two: Contrast Enhancement

By (3) we may make the object look brighter; however, the contrast of the object is still low. Therefore, we need to adopt a contrast stretching technique to increase

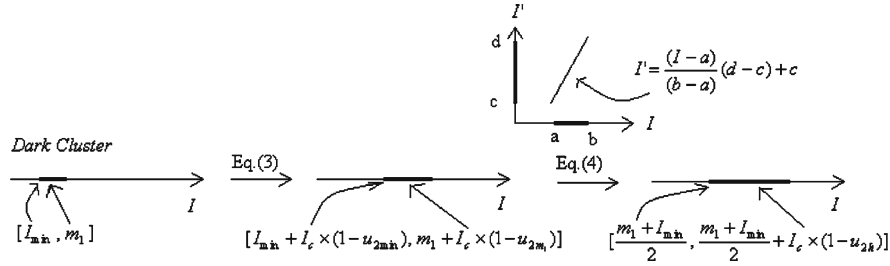


Figure 4. An illustration of the effect of Stage 2 and Stage 3.

the dynamic range of the I component of the object. Figure 4 illustrates such an enhanced procedure. For each pixels of which I component satisfies the condition, $I_j \leq m_1 + I_c \times (1 - u_{2m_1})$, we use the following equation to enhance its contrast

$$I_j = \frac{\{I_j - [I_{\min} + I_c \times (1 - u_{2\min})]\} \times [I_c \times (1 - u_{2h}) + 0.5m_1 + 0.5I_{\min}]}{[m_1 + I_c \times (1 - u_{2m_1})] - [I_{\min} + I_c \times (1 - u_{2\min})] + 0.5m_1 + 0.5I_{\min}} \quad (4)$$

where u_{2h} represents the degree of the gray level $\frac{m_1 + I_{\min}}{2}$ belonging to the bright cluster. Via (4), the brightness interval, $[I_{\min} + I_c \times (1 - u_{2\min}), m_1 + I_c \times (1 - u_{2m_1})]$, is stretched to be the interval, $[\frac{m_1 + I_{\min}}{2}, \frac{m_1 + I_{\min}}{2} + I_c \times (1 - u_{2h})]$ in order to enhance the dynamic range of the dark cluster.

After each pixel's I component has been compensated we transform the HIS color model back to the RGB color model to display the image.

3. Brief Introduction to AISNFS

In our previous work [21], we have developed a training algorithm for applying AISNFSs in solving pattern recognition problems. Here we will introduce how to train AISNFSs for solving function approximation problems. The immune system is a highly complicated system. Many properties of immune systems attract a great amount of attentions from compute scientists and engineers. A rich source of mechanisms in immune systems may serve as metaphors for designing computational systems. Different artificial immunes systems are inspired by different subsets of the available metaphors.

Basically, the architecture of the proposed AISNFS, shown in Figure 5, resembles the architecture of the RBF network or its variants [22–25]. The major differences between the AISNFS and these well-known networks are as follows. Generally, the RBF network or its variants are trained by the *supervised learning* such as orthogonal least squares, backpropagation algorithm, etc. However, our AISNFS is trained in an *on-line learning* environment. In addition, we introduce the strength parameter, s_j , to represent the strength of the j th rule. Owing to the appearance of the strength

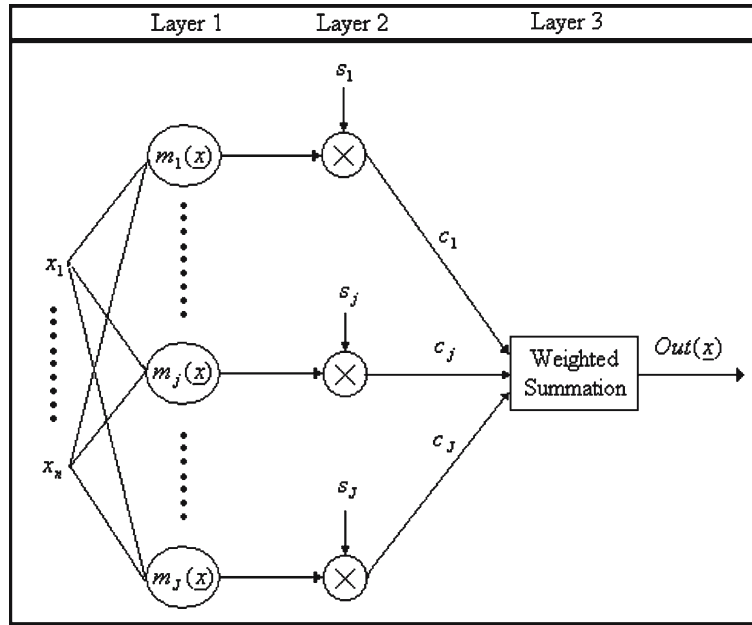


Figure 5. The architecture of the AISNFS for function approximation problems.

parameter, the AISNFS is different to the RBF network or its variants in the way of the computation of the network's outputs.

The algorithm for training an AISNFS for solving function approximation problems is given as follows:

- Step 1:* Set $J = 0$, decide the value of the initial strength s_0 , and initialize the parameters, $\eta_w, \eta_\sigma, \eta_s$ and η_c , where J is the number of rules in AISNFS, $\eta_w, \eta_\sigma, \eta_s$ and η_c indicate the learning rates.
- Step 2:* Present an input pattern $\underline{x}$ into the system. Suppose the desired output of the input pattern is d .
- Step 3:* If there is no neuron in the system then go to Step 8 to generate a new neuron to respond to the first data pattern $\underline{x}$. Otherwise, compute the outputs of the neurons in the present system using the following equation:

$$m_j(\underline{x}) = \exp \left\{ - \sum_{i=1}^n \left(\frac{x_i - w_{ji}}{\sigma_{ji}} \right)^2 \right\} \quad (5)$$

- Step 4:* Find the winner j^* by selecting the neuron with the largest similarity which is computed by multiplying the neuron's output, $m_j(\underline{x})$, with its corresponding strength, s_j :

$$s_{j^*}m_{j^*}(\underline{x}) = \text{Arg} \max_{j=1, \dots, J} (s_j m_j(\underline{x})) \quad (6)$$

Here we assume the present number of neurons in the system is J .

Step 5: Check whether the similarity between the winner and the input pattern $\underline{x}$ is larger than the pre-specified threshold θ :

$$s_{j^*}m_{j^*}(\underline{x}) \geq \theta \quad (7)$$

If this condition is met, then find the voting set, V_s , consisting of neurons of which similarities are also larger than the threshold θ . Otherwise, go to Step 8 to generate a new neuron.

Step 6: Compute the system output. The system output is computed by the weighted summation method as follows:

$$\text{Out}(\underline{x}) = \frac{\sum_{j \in V_s} c_j m_j(\underline{x})}{\sum_{j \in V_s} m_j(\underline{x})} \quad (8)$$

That is, all neurons in the voting set will cooperate to compute the network's outputs. The weighting factor for each neuron is provided by the value of membership function, $m_j(\underline{x})$.

Step 7: Compute the difference between the system output and the desired output, $e = |\text{Out}(\underline{x}) - d|$. Then check whether the difference is smaller than the threshold, θ_{error} , where θ_{error} is a positive real-valued constant pre-specified by the user. If yes, update the winner's strength in the following way:

$$s_{j^*}(n+1) = s_{j^*}(n) + \eta_s \times \frac{1}{f_{j^*}} \times s_{j^*} \times m_{j^*}(\underline{x}) \quad (9)$$

Otherwise, we update the winner's weight vector, strength and its output in the following way. For the case $d - \text{Out}(\underline{x}) > \theta_{\text{error}}$, the values of $\underline{w}_{j^*}(n)$, $\sigma_{j^*}(n)$, and $c_{j^*}(n)$, are adjusted by an amount proportional to the positive of their corresponding gradients at their current values and the value of $s_{j^*}(n)$ is adjusted by an amount proportional to the negative of its gradient at its current value. For the case $d - \text{Out}(\underline{x}) < -\theta_{\text{error}}$, the values of these four parameters are all adjusted by an amount proportional to the negative of their gradients at their current values. Then Go to Step 2.

Step 8: Generate a new neuron whose weight vector and strength are initialized as follows:

$$\underline{w}_{J+1} = \underline{x} \quad (10)$$

$$\sigma_{J+1} = \sigma_0 \quad (11)$$

$$c_{J+1} = d \quad (12)$$

$$s_{J+1} = s_0 \quad (13)$$

$$J = J + 1 \quad (14)$$

Then Go to Step 2.

Note that at the end of epoch, we will delete the neurons of which strengths are smaller than the initial value s_0 . To improve the training performance, we may continue the training procedure until a pre-specified number of epochs is reached. After the network has been trained, we proceed to the testing mode. In the testing mode, the trained network can be used to predict the output when an input pattern, $\underline{x}$, is presented to the network. First, we find the voting set. If the set is not an empty set then the output is computed by the weighted summation method. Otherwise, we use the winner's output, c_{j^*} , as the system output.

4. Experimental Results

We apply the proposed compensation algorithm to the backlight images collected by us to demonstrate its performance. We collected 50 images of which sizes are all 640×480 for various kinds of objects under different lighting conditions. The image data set consists of 45 backlight images and 5 normal images. We then manually determined the optimal amount of compensation for each backlight image. One thing should be emphasized is that the manually chosen optimal amount of compensation for each backlight image was decided by a process of trial and error. That is, we continued to use a different amount of compensation to compensate an image until the quality of the compensated image was satisfactory. We then split the data set into a training data set and a testing data set. The training data set consists of 22 backlight images and 3 normal images. The testing set consists of the remaining 25 images. We used the training data set to train a two-layer perceptron with the backpropagation algorithm, a two-layer RBF network, and an AISNFS with the same number of hidden nodes. The results are tabulated in Table 1. Obviously, the trained AISNFS outperformed the other two networks based on the comparison of the root mean square error (RMSE). Since the trained AISNFS achieved the lowest RMSE we decided to use the trained AISNFS to continue the experiments.

First, we used the proposed compensation algorithm incorporated with the trained AISNFS to compensate the whole 50 images. Then for the comparison purpose, these 50 images were also compensated by using the histogram equalization algorithm in the Photoshop software package. Methods for assessing image quality can be divided into two categories: objective and subjective. While objective methods adopt some kinds of mathematical criteria to measure image quality, subjective methods are based on the perception of a group of human

Table 1. The performance of the three kinds of function approximators.

Approximator	Architecture	Training set RMSE	Testing set RMSE
AISNFS	$4 \times 15 \times 1$	9.32	11.11
RBF	$4 \times 15 \times 1$	11.35	11.54
MLP	$4 \times 15 \times 1$	11.01	11.92

viewers. In objective methods, the quality of an image is usually estimated by comparison with a known reference image. The root-mean-square error (rms) and signal-to-noise ratio (SNR) are two popular criteria for measuring image quality. Although objective methods offer a simple and objective mechanism for measuring image quality we do not adopt them to measure the quality of compensated images. The reasons are as follows. First, compensated images ultimately are viewed by human beings; therefore, measuring the quality of compensated images by the subjective evaluations of a human observer is more appropriate. Second, we are unable to have the reference images for computing the measures. Thus we adopted the subjective assessment method to evaluate the performance of the histogram equalization algorithm and our proposed compensation algorithms. As the assessment measure, the five level grading scales shown in Table 2 were used. Ten subjects were asked to evaluate the compensated images. Table 3 tabulates the statistics of the votes for these two algorithms. Based on the votes we found that our compensation algorithm received a better grade than the histogram equalization algorithm.

Some of the images compensated by the proposed algorithm and the histogram equalization are shown in Figure 2(d). Their histograms are shown in Figure 6. To see these histograms suddenly, one may make an erroneous judgment that the histogram-equalized images will definitely more satisfactory than those images compensated by our algorithm. The erroneous judgment may come from the following observations. Observing the histograms, some common features exist. For all images, our algorithm gives similar shaped histogram to the original images. On the contrary, the histogram equalization algorithm produces histograms that are completely different from the original histograms. The resultant images have almost flat shapes for the dark regions and different shaped histograms for the bright regions. However, by viewing the enhanced images shown in Figure 2(d) we will make an opposite judgment. One may find both our algorithm and the histogram equalization algorithm have greatly improved the main objects in these backlight images. However, we found that in some areas (e.g. especially the backgrounds) of the histogram-equalized images the color information was seriously distorted. On the contrary, images compensated by our algorithm do not exhibit the distortion effect. Therefore, our algorithm performed superior compensation to the histogram equalization algorithm based on the comparisons of these 50 compensated images.

Table 2. The five level grading scales.

Scale	Rating words
1	Bad
2	Poor
3	Fair
4	Good
5	Excellent

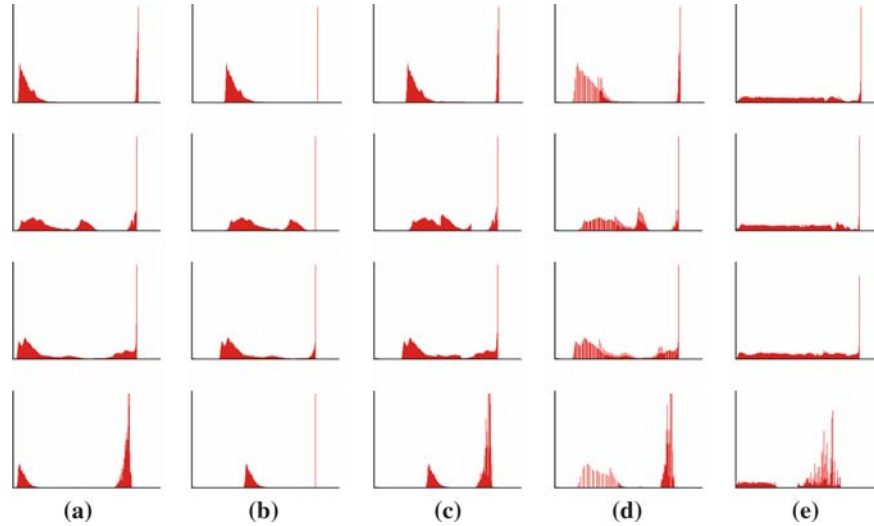


Figure 6. Histograms of Figure 2. (a) The left column corresponds to the original images, (b) the second column corresponds to the first simple method of adding the same compensation amount into each pixel, (c) the central column corresponds to the second simple method of adding the same compensation amount into each pixel belonging to the main objects, (d) the fourth column corresponds to the proposed algorithm, (e) the right column corresponds to the histogram-equalized images.

Another appealing property of our compensation algorithm is that the algorithm can be improved on-line. To demonstrate the on-line learning property we carried out another experiment. We selected five images, which received the lowest grades, from the testing set. Then we directly used these five chosen images to re-train the AISNFS. One thing should be emphasized is that we did not combine these five image into the original training data set to form a larger training data set. After training, the 50 images were compensated by the compensation algorithm incorporated with the newly trained AISNFS. The performance was tabulated in Table 4. Although the average score just increased a very small amount the RMSE decreased more apparently. This experiment demonstrated that the performance of our algorithm could be improved on-line. If the user is not satisfied with the compensated image achieved by the algorithm incorporated with a trained AISNFS he or she may decide the amount of compensation by him or herself to compensate the image. Then he or she can use this newly compensated image to re-train the originally trained AISNFS to further improve the performance. As the process continues the performance of the compensation algorithm will become more fit to the user's personal requirements.

5. Conclusions

In this paper, we propose a new compensation algorithm for backlight images. We introduce four features to represent the information of the brightness histogram of a backlight image and use an AISNFS to determine the amount of compen-

Table 4. The performance of the on-line learning capability.

Performance	RMSE	Average score
Original trained AISNFS	10.22 (= (9.32 + 11.11)/2)	3.65
Newly trained AISNFS	8.16	3.66

sation. Instead of adding the same amount of compensation to each pixel, we add different amount of compensation to each pixel based on its degree of belonging the bright region. The algorithm can be used to compensate backlight images in a software way. In addition, the algorithm does not require very complex computations and hardware, so it may be implemented in an electronic camera and then used to compensate images. Another appealing property of our compensation algorithm is that the algorithm can be improved on-line.

Acknowledgement

This work was supported by the MOE Program for Promoting Academic Excellent of Universities under the grant number EX-91-E-FA06-4-4. References, the National Science Council, Taiwan, R.O.C, under the NSC 93-2524-S-008-002-, 94-2213-E-008-032-, and 93-2213-E-008-003-, and the Ministry of Economic Affairs under the 95-EC-17-A-02-S1-029.

References

1. Morimura, A., Uomori, K., Kitamura, Y., Fujioka, A., Harada, J., Iwamura, S. and Hirota, M.: A digital video camera system, *IEEE Trans. Consumer Electron.*, **36**(4) (1990), 866–875.
2. Haruki, T. and Kikuchi, K.: Video camera system using fuzzy logic, *IEEE Trans. Consumer Electron.*, **38**(3), (1992), 624–634.
3. Murakami, M. and Honda, N.: An exposure control system of video cameras based on fuzzy logic using color information, *Proceedings of the Fifth IEEE International Conference on Fuzzy Systems*, vol. 3, 1996.
4. Shimizu, S., Kondo, T., Kohashi, T., Tsurata, M. and Komuro, T.: A new algorithm for exposure control based on fuzzy logic for video cameras, *IEEE Trans. Consumer Electron.*, **38**(3), (1992), 617–623.
5. Gonzalez, R. C. and Woods, R. E.: *Digital Image Processing*, Addison-Wesley, MA, 1992.
6. Zimmerman, J., Pizer, S., Staab, E., Perry, E., McCartney, W. and Brenton, B.: Evaluation of the effectiveness of adaptive histogram equalization for contrast enhancement, *IEEE Trans. Med. Imaging* (1988), 304–312.
7. Kim, Y. T.: Contrast enhancement using brightness preserving bi-histogram equalization, *IEEE Trans. Consumer Electron.*, **43**(1), (1997), 1–8.
8. Kim, T. K., Paik, J. K. and Kang, B. S.: Contrast enhancement system using spatially adaptive histogram equalization with temporal filtering, *IEEE Trans. Consumer Electron.*, **44**(1), (1998), 82–86.
9. Boccignone, G.: A multiscale contrast enhancement method, In: *Proceedings of the International Conference on Image Processing*, pp. 306–309, 1997.

10. Caselles, V., Lisani, J. L., Morel, J. M. and Sapiro, G.: Shape preserving local contrast enhancement, In: *Proceedings of the International Conference on Image Processing*, pp. 314–317, 1997.
11. Sakaue, S., Tamura, A., Nakayama, M. and Maruno, S.: Adaptive gamma processing of the video cameras for the expansion of the dynamic range, *IEEE Trans. Consumer Electron.*, **41**, (1995), 555–562.
12. Ketcham, D. J., Lowe, R. and Weber, W.: Seminar on image processing, In: *Real-Time Enhancement Techniques*, Hughes Aircraft, pp. 1–6, 1976.
13. Hummel, R.: Image enhancement by histogram transformation, *Comp. Graph. Image Process.*, **6**, (1977), 184–195.
14. Tom, V. T. and Wolfe, G. J.: Adaptive histogram equalization and its applications, *SPIE Applicat. Dig. Image Process.* **359**, (1982), 204–209.
15. Pizer, S. M., Amburn, E. P., Austin, J. D., Cromartie, R., Geselowitz, A., Greer, T., Romeny, B.H., Zimmerman, J.B. and Zuiderveld, K.: Adaptive histogram equalization and its variations, *Comp. Vis. Graph. Image Process.* **39**(3), (1987), 355–368.
16. Stark, J. A.: Adaptive image contrast enhancement using generalizations of histogram equalization, *IEEE Tran. Image Processing* **9**(5), (2000), 889–896.
17. Kim, J. Y., Kim, L. S. and Hwang, S.H.: An advance contrast enhancement using partially overlapped sub-block histogram Equalization, *IEEE Trans. Circuits Syst. Video Technol.* **11**(4), (2001), 475–484.
18. Su, M. C., Guo, J. H., Lin, D. -T. and Wang, G. C.: New compensation algorithm for color backlight images, In: *IEEE International Conference on Neural Networks*, Hawaii, USA, pp. 1396–1400, 2002.
19. Bezdek, J. C.: *Fuzzy Mathematics in Pattern Classification*, Ph. D Thesis, Cornell University, 1973.
20. Dunn, J. C.: A fuzzy relative of the ISODATA process and its use in detecting compact, well separated clusters, *J. Cybern.* **3**(3) (1973), 35–57.
21. Su, M. C., Yang, Y. S., Chou, C. H., Lai, E. and Hsiao, M. N.: An on-line learning neuro-fuzzy system based on artificial immune systems, *2004 IEEE International Joint Conference on Neural Networks IJCNN*, Hungary, pp. 1073–1078, 25–29 July, 2004.
22. Moody, J. and Darken, C. J.: Fast learning in networks of locally tuned processing units, *Neural Comput.*, **1** (1989), 181–194.
23. Wang, L. X. and Mendel, J. M.: Fuzzy basis functions, universal approximation, and orthogonal least-squares learning, *IEEE Trans. Neural Networks* **3**(5), (1992), 807–814.
24. Kim, H. M. and Mendel, J. M.: Fuzzy basis functions: comparisons with other basis functions, *IEEE Trans. Fuzzy Syst.*, **3**(2), (1995), 158–168.
25. Le, C. W. and Shin, Y. C.: Construction of fuzzy basis function networks using adaptive least squares method, *IFSA World Congress and 20 th NAFIPS International Conference*, pp. 2630–2635, 2001.