A SKIN DETECTOR BASED ON NEURAL NETWORK

Li Chen[1], *Jiliu Zhou*[1], *Zhiming Liu*[1], *Wei Chen*[1], *Guoqing Xiong*[2]
[1]College of Electronic Information,
Sichuan University, Chengdu, 610064, P.R.China.
[2]Telecommunication Station, West China Medical School,
Sichuan University, Chengdu, 610064, P.R.China.

Abstract: A large body of human image processing techniques use skin detection as a first step for subsequent feature extraction. The objective of this work is to provide an efficient tool to detect human skin in color images. Well-established methods of color modeling, such as histograms and Gaussian mixture models have enabled the construction of suitably accurate skin detectors. However such techniques are not ideal for use in various environments. We describe a method of skin detection using a back propagation neural network, and show considerable good performance for a large variety of color images. We also introduce genetic algorithms into the weights and biases optimization of the neural network. The paper will focus on the novel approach to design a neural network based skin detector, which will be later used to retrieve skin-like homogeneous regions in color face images.

Keywords: Skin detector, Neural network, Back propagation, Genetic Algorithms,

1. Introduction

Human skin color has been proven to be a powerful fundamental cue and used for many applications ranging from face detection and tracking [1] to intelligent human-computer interfaces. In order to make use of skin color information, much research has been directed to understanding its property. The analysis shows that human skin colors are distinctive from the color of other natural objects, and fall in a small region in the chrominance plane. Several studies have shown that the major skin color difference between different people lies largely in their intensity rather than in their chrominance [2]. Thus, if the image is first converted into a color space, which provides a separation into a luminance channel and two chrominance components like the normalized (r, g, b) color space, and then skin-like regions can easily be detected. A majority of the skin detection algorithms use color histograms for segmentation, either direct or for ML estimation. Others perform pixel classification based on predefined ranges in color space. A color histogram model can be built as demonstrated by Michael *et al.* [3], if we have a large body of skin pixels set. Researchers also find that the skin color distribution can be well modeled by a mixture of Gaussians [4][5]. Using skin color as a cue to detect a face has several advantages: Firstly, Skin detection techniques can be both simple and accurate; Under white lighting conditions color does not vary significantly with orientation or view angles.

However, color is not a physical phenomenon. It is a perceptual phenomenon that is related to the spectral characteristics of electro-magnetic radiation in the visible wavelengths striking the retina [2]. Neural networks are parameterized non-linear models used for empirical regression and classification modeling. Their flexibility makes them able to discover more general relationships in data than traditional statistical models. Unsurprisingly we find good performance in applying neural networks to skin detection. Further more, an evolutionary search procedure (GAs) also been introduced into the neural networks training process and results show much better performance.

2. Back Propagation Networks

Back Propagation (BP) Neural Network simulates a back propagation algorithm that is a well-known algorithm widely used in artificial intelligence. A BP network consists of at least three layers of units: an input layer, at least one intermediate hidden layer, and an output layer.

Li Chen E-mail:holympus@163.com

0-7803-7547-5/02/$17.00 ©2002 IEEE

Typically, units are connected in a feed-forward fashion with input units fully connected to units in the hidden layer and hidden units fully connected to units in the output layer.

When a BP network is cycled, an input pattern is propagated forward to the output units through the intervening input-to-hidden and hidden-to-output weights. The neurons are interconnected in such a way that information relevant to the I/O mapping is stored in the weights.

With BP networks, learning occurs during a training phase in which each input pattern in a training set is applied to the input units and then propagated forward. The pattern of activation arriving at the output layer is then compared with the correct (associated) output pattern to calculate an error signal. The error signal for each such target output pattern is then back propagated from the outputs to the inputs in order to appropriately adjust the weights and biases in each layer of the network. After a BP network has learned the correct classification for a set of inputs, it can be tested on a second set of inputs to see how well it classifies untrained patterns.

We can interpret the output of a BP network as a classification decision and explore how the BP network can be used to detect human skin information and classify a pixel belongs to skin or non-skin. There are two issues that must be addressed in design of a BP networks-based skin detector, the choice of the key skin features and the construction of the neural networks. As we have discussed before, human skin colors can be distinctive from most other objects' colors. So the first step is to convert a pixel into a suitable color space. Several color spaces have been utilized to label pixels as skin including RGB, normalized RGB, HSV(or HSI), YIQ etc. In our work, we use normalized RGB color space which is a color model commonly used in face detection; it reduces the sensitivity to illumination changes while staying very close to the 'usual' $(r; g; b)$:

$$\begin{cases} r = \dfrac{R}{R+G+B} \\ g = \dfrac{G}{R+G+B} \\ b = \dfrac{B}{R+G+B} \end{cases} \tag{1}$$

it can been seen that r+g+b=1. The normalized colors can be effectively represented using only *r* and *g* values as *b* can be obtained by noting b=1-r-g.

After converting a pixel into the Normalized RGB color space, we get a color feature of two-dimension vector (r, g) by eliminating the influence of luminance as input. The output of the BP networks has only one unit indicating that the pixel belongs to skin (output=1) or non-skin (output=0) areas. Taking account both of training time and ability of classifying, we adopt construction of BP networks as follows:

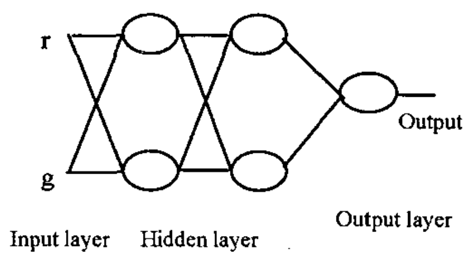

Fig.1 skin detector based on neural networks

The neural network is composed of 2 input neurons (representing the characteristics of pixel's chrominance components), 4 hidden neurons in 2 layers, and 1 neuron as output unit. The number of neurons in hidden layers is obtained by the experimental formula [6]:

$$n_i = \sqrt{n+m} + \alpha \tag{2}$$

where n and m are the number of input and output neuron respectively. α is a constant between 1 and 10.Each neuron contains the weighted sum of its inputs filtered by a sigmoidal (S-shaped) transfer function:

$$f(x) = \frac{1}{1+e^{\alpha x}} \quad (3).$$

The parameter σ plays a very important role in the convergence of the neural networks: the larger σ is, the neural networks will converge more quickly, but also easy get unstable. On the other hand, if σ is too small, the convergence of the neural networks will be time consuming though may get good result. After applying GAs to train the neural network, the optimization requires improvement of the neural network, so σ can be a relative small one.

3. Hybrid Training Algorithms

Back-propagation is a gradient descent search algorithm, which tries to minimize the total square error between actual output and target output of a neural networks. This error is used to guide BP's search in the weight and bias space. There have been some successful applications of BP algorithms. However, drawbacks with the BP algorithms do exist due to its gradient descent nature. Studies show back propagation training algorithms is very sensitive to initializing conditions and often get trapped in local minimum of the function. To overcome those drawbacks, global search procedures like genetic algorithms (GAs) can be introduced effectively into the training process.

In our work, we first optimize weights and biases of the BP networks by using genetic algorithms, and then apply back propagation algorithms to refine the neural networks since back propagation algorithm is good at searching for the optimum around the initialized area. In such a way we can make fully use of the advantages of both algorithms.

3.1 Genetic algorithms based optimization

The first step of applying GAs to training a neural network is to encode the solutions. In our work, a solution is composed of 15 parameters representing 10 weights and 5 biases for the neural networks. We adopt real number strings to encode the possible solutions:

$$s = (w_1, \ldots, w_i, \ldots, w_{10}, b_1, \ldots, b_j, \ldots, b_5)$$

For each solution, we input the training set into the neural network, and calculate the overall system error E:

$$E = \sum_m \left(Y_m - \overline{Y_m}\right)^2 \tag{3}$$

E also can be seen as energy function of the networks. Y_m and $\overline{Y_m}$ represent the training data m's expected output and the actual output respectively. Fitness function is defined as follows:

$$Fitness = \frac{1}{1+E} \tag{4}$$

GAs adopts elitist model selection, arithmetic crossover [7] and non-uniform mutation.

As for crossover operation, we take two selected parentsS_1, S_2 for example, and get the following two offspring:

$$\begin{cases} S_1' = aS_1 + (1-a)S_2 \\ S_2' = aS_2 + (1-a)S_1 \end{cases} \qquad 0 \le a \le 1$$

As for mutation operation, we choose according to mutation probability (P_m) a solution ($S=(s_1, s_2, \ldots, s_i, \ldots, s_{15})$) and its gene s_i ($s_i \in (Li, Ui)$). Generate a integrate random (random $\in \{0,1\}$), the result is $S'=(s_1, s_2, \ldots, s_i', \ldots, s_{15})$ ($i \in \{1, \ldots, 15\}$)

$$si' = \begin{cases} s_i + \Delta(t, Ui - s_i) & \text{random=0,} \\ s_i - \Delta(t, s_i - Li) & \text{random=1.} \end{cases}$$

$\Delta(t, y) = y(1 - r^{(1-\frac{t}{T})^b})$, in which r is a random belongs to the range of [0,1], T the max evolutionary generation, t evolutionary generation and b system parameter.

3.2 Back propagation algorithm based refinement

We define system error E_{total}:

$$E_{total} = \frac{1}{2}\sum_{m}\left(Y_m - \overline{Y}_m\right)^2 \quad (5)$$

The BP learning process begins with the neural networks trained by Genetic algorithms. The training set is applied to the network, and the network produces outputs based on the current state of its synaptic weights. Accumulate the total error of network works and adjust the weights and biases in proportion to the negative of an error derivative with respect to each weight and bias:

$$\begin{cases}\Delta w_{ji} = -\varepsilon\left(\partial E_{total} / \partial w_{ji}\right) \\ \Delta \theta_i = -\varepsilon\left(\partial E_{total} / \partial \theta_i\right)\end{cases} \quad (6)$$

where, ε is a small iterative step which can also influence whether the network achieves a stable solution. Weights and biases move in the direction of steepest descent on the error surface defined by the total error. The training process terminates when E_{total} satisfies the required precision.

4. Skin-like Regions Detection

In applying the trained network to detect the skin-like color pixels, we find it very important to introduce a coarse selection into the skin-like color detection procedure. The proposed algorithm is then realized as the following steps.

1. Convert a pixel into normalized RGB color space. If r, g, b satisfy the simple criterion (r>g) or (r>b), put the pixel into the trained neural networks, otherwise, continue step 1 and process the next pixel.

2 According to the corresponding output, classify the pixel into skin-like pixel or non-skin-like.

3. Accomplish the detection of the skin-like color regions.

By adopting a simple coarse selection in step 1, much time has been saved, generally 80%. For example, detecting skin-like pixels in a color image from BIOID database needs only about 80ms against 400ms without data preprocessing.

5. Results And Conclusion

Test image set is composed of one hundred color images, including one-face color images from BIOID dataset and images taken in our lab or some web images containing two or more faces.

The results show the BP neural networks based skin detector can detect skin-like pixels more accurately under adverse illumination conditions. As compared to the results of some traditional methods, the neural network trained by proposed hybrid algorithms performs much better, especially in difficult conditions. It is also impressive to find that the training processing is affected much by training algorithm. A comparison is made with the corresponding results obtained with conventional back propagation algorithms and that we presented in section 3. The traditional back propagation algorithm training sometimes gets into local optimum and the performance is not always good. On the other hand, the network trained by hybrid algorithm provides us with a more robust skin detector. Some results are shown in Fig 3 and Fig 4.

References

[1] M. Hunke and A. Waibel. "Face locating and tracking for human-computer interaction". In *Proc. Twenty-Eight Asilomar Conference on Signals, Systems & Computers*, Monterey, CA, USA, 1994.

[2] J. Yang and A. Waibel. "A real-time face tracker". In *Proceedings of the Third Workshop on Applications of Computer Vision*, pp 142-147,1996.

[3] Michael J. Jones James M. Rehg "Statistical Color Models with Application to Skin Detection". *In Computer Vision and Pattern Recognition (CVPR 99), Ft. Collins, CO*, pp 274-280, June, 1999.

[4] J. Yang, W. Lu, and A. Waibel. "Skin-color modeling and adaptation". *In Proc. ACCV*, pp. 687–694, 1998

[5] S.J McKenna, S. Gong, Y. Raja, "Modelling facial colour and identity with Gaussian mixtures", *Pattern Recognition*, 31(12)(1998), pp. 1883–1892.

[6] Liming Zhang. *Models and Applications of Artificial Neural Networks*. Fudan Univerdsity Press. 1993

[7]. Michalewicz Z, Janikow C Z, Krawczyk JB. "A modified genetic algorithm for optimal control problems". *Computer Math Appli*, 23 (12) ,pp 83- 94, 1992.

Fig 3: (a) original image (b) Gauss model (c) NN (BP) (d) NN (hybrid)

origional image gauss mode BP(hybrid)

Fig 4: Skin detection are performed using the network trained on 25088 normalized RG skin samples