Volume 1, No. 5, December 2010
Journal of Global Research in Computer Science
ISSN-2229-371X
RESEARCH PAPER
Available Online at www.jgrcs.info

PRE PROCESSING HIERARCHY DESIGNED FOR ENHANCED FACE DETECTION

S. Venkatesan*[1] and Dr. Srinivasa Rao madane[2]

[1]Ph.D., Scholar, Department of Computer Science and Engineering
Anna University Coimbatore, Coimbatore, India
selvamvenkatesan@gmail.com

[2]Professor & Principal, Department of Computer Science and Engineering
Priyadarshini college of Engineering, vaniyambadi, India
raomadane@gmail.com

Abstract: Face recognition has been grown as a prime security idea since last decade. Face detection is the basic step in face recognition. In this paper, we have discussed the basic pre processing steps for face detection 2D color images with single or multiple faces. Proposed steps work in various steps including skin color segmentation, morphological operations. The beauty of this algorithm is that it is scale independent and orientation invariant. This pre processing steps is checked against various images with dynamic condition, which shows accuracy range of 90% to 100 %.

Keywords: Filtering, Hue, Saturation, Connected Image Smoothing, Histogram Morphological operations

INTRODUCTION

Face detection is concerned with finding whether or not there are any faces in a given image (usually in gray scale) and, if present, return the image location and content of each face. This is the first step of any fully automatic system that analyzes the information contained in faces (e.g., identity, gender, expression, age, race and pose). While earlier work dealt mainly with upright frontal faces, several systems have been developed that are able to detect faces fairly accurately with in-plane or out-of-plane rotations in real time.

Most detection systems carry out the task by extracting certain properties (e.g., local features or holistic intensity patterns) of a set of training images acquired at a fixed pose (e.g., upright frontal pose) in an off-line setting. To reduce the effects of illumination change, these images are processed with histogram equalization or standardization (i.e., zero mean unit variance).Based on the extracted properties, these systems typically scan through the entire image at every possible.

In our research we are doing the following basic operations, which are considered to be pre-processing steps. Operations that are fundamental to any image analysis is Operations based on filtering Operations based on image smoothing Operations based on image histogram on mathematical morphology on simple mathematics.

FILTERING AND THRESHOLD ANALYSIS

Various noisy sources may exist in the image. The fine details of the image represent high frequencies, which mix up with those of noise. One kind of noise, which occurs in all recorded images to a certain extent, is detector noise. This kind of noise is due to the discrete nature of radiation, i.e. the fact that each imaging system is recording an image by counting photons. A common form of noise is data dropout noise (commonly referred to as intensity spikes, speckle or salt and pepper noise). Here, the noise is caused by errors in the data transmission. The corrupted pixels are either set to the maximum value (which looks like snow in the image) or have single bits flipped over. In some cases, single pixels are set alternatively to zero or to the maximum value, giving the image a `salt and pepper' like appearance. The noise is usually quantified by the percentage of pixels, which are corrupted. Low pass filtering, otherwise known as "smoothing", is employed to remove high spatial frequency noise from a digital image. Noise is often introduced during the analog-to-digital conversion process as a side effect of the physical conversion of patterns of light energy into electrical patterns. There are several common approaches to removing this noise: If several copies of an image have been obtained from the source, some static image, then it may be possible to sum the values for each pixel from each image and compute an average. This is not possible, however, if the image is from a moving source or there are other time or size restrictions. If such averaging is not possible, or if it is insufficient, some form of low pass spatial filtering may be required.

There are two main types:

a) Reconstruction filtering : Here an image is restored based on some knowledge of the type of degradation it has undergone.

b) Enhancement filtering: It attempts to improve the (subjectively measured) quality of an image for human or machine interpretability. The above filters are all space invariant in that the same operation is applied to each pixel location. A non-space invariant filtering, using the above filters, can be obtained by changing the type of filter or the weightings used for the pixels for different parts of the image. Non-linear filters also exist which are not space invariant; these attempt to locate edges in the noisy image before applying smoothing, a difficult task at best, in order to reduce the blurring of edges due to smoothing. The ***median filter*** is a simple edge-preserving smoothing filter. It may be applied prior to segmentation in order to reduce the amount of noise in a stack of 2D images. The filter works by sorting pixels covered by a N x N mask according to their gray value. The *center pixel* is then replaced by the *median of these pixels*, i.e., the middle entry of the sorted list So low pass filters are used to obliterate some details in the image. In this experiment prewitt filtering is used to suppress the noise. Under normal conditions the facial feature possesses relatively lower gray level and the intensity histogram of the face image produces the shape of the twin peaks. One peak corresponds to the lighter parts of the face such as cheeks; forehead etc. and other peak correspond to the darker parts of the face. Then the threshold value should be chosen in such a way that the facial features become distinct with respect to the lighter parts of the face.

Threshholding

In many vision applications, it is useful to be able to separate out the regions of the image corresponding to objects in which we are interested, from the regions of the image that correspond to background. Thresholding often provides an easy and convenient way to perform this segmentation on the basis of the different intensities or colors in the foreground and background regions of an image. In addition, it is often useful to be able to see what areas of an image consist of pixels whose values lie within a specified range, or band of intensities (or colors). The input to a thresholding operation is typically a grayscale or color image. In the simplest implementation, the output is a binary image representing the segmentation. Black pixels correspond to background and white pixels correspond to foreground (or vice versa). In simple implementations, the segmentation is determined by a single parameter known as the intensity threshold. In a single pass, each pixel in the image is compared with this threshold. If the pixel's intensity is higher than the threshold, the pixel is set to, say, white in the output. If it is less than the threshold, it is set to black. Thresholding produces a segmentation that yields all the pixels that, in principle, belong to the object or objects of interest in an image. An alternative to this is to find those pixels that belong to the borders of the objects. Techniques that are directed to this goal are termed edge-finding techniques. Representing an image by its edges has the further advantage that the amount of data is reduced significantly while retaining most of the image information. In practice, edge detection is performed in the spatial domain, because it is computationally less expensive and often yields better results. Since edges correspond to strong illumination gradients, we can highlight them by calculating the derivatives of the image. The gradient has a large peak centered around the edge, the edge has become "thick" due to the thresholding. The edge occurs at the peak, we can localize it by computing the laplacian (in one dimension, the second derivative with respect to t) and finding the zero crossings.

IMAGE SMOOTHING

These algorithms are applied in order to reduce noise and/or to prepare images for further processing such as segmentation. We distinguish between linear and non- linear algorithms.

a) Uniform filter - The output image is based on a local averaging of the input filter where all of the values within the filter support have the same weight.

b) Triangular filter - The output image is based on a local averaging of the input filter where the values within the filter support have differing weights. In general, the filter can be seen as the convolution of two (identical) uniform filters either rectangular or circular and this has direct consequences for the computational complexity.

c) Gaussian filter - smoothes or blurs an image by performing a convolution operation with a Gaussian filter kernel. The text fields labeled kernel size allow you to change the size of the convolution kernel in each dimension. A value of 3 denotes a 3x3 kernel. Odd values are required.

HISTOGRAM EQUALIZATION

Histogram modeling techniques (e.g. histogram equalization) provide a sophisticated method for modifying the dynamic range and contrast of an image by altering that image such that its intensity histogram has a desired shape. Unlike contrast stretching, histogram-modeling operators may employ non-linear and non-monotonic transfer functions to map between pixel intensity values in the input and output images. Histogram equalization employs a monotonic, non-linear mapping which re-assigns the intensity values of pixels in the input image such that the output image contains a uniform distribution of intensities (i.e. a flat histogram). This technique is used in image comparison processes (because it is effective in detail enhancement) and in the correction of non-linear effects introduced by, say, a digitizer or display system.

MORPHOLOGICAL OPERATIONS

Morphology is a broad set of image processing operations that process images based on shapes. Morphological operations apply a structuring element to an input image, creating an output image of the same size. In a morphological operation, the value of each pixel in the output image is based on a comparison of the corresponding pixel in the input image with its neighbors. By choosing the size and shape of the neighborhood, you can construct a morphological operation that is sensitive to specific shapes in the input image.

The most basic morphological operations are dilation and erosion. Dilation adds pixels to the boundaries of objects in an image, while erosion removes pixels on object boundaries. The number of pixels added or removed from the objects in an image depends on the size and shape of the structuring element used to process the image. In the morphological dilation and erosion operations, the state of any given pixel in the output image is determined by applying a rule to the corresponding pixel and its neighbors in the input image. The rule used to process the pixels defines the operation as dilation or erosion. This table lists the rules for both dilation and erosion.

Rules for Dilation and Erosion :

Operation	Rule
Dilation	The value of the output pixel is the maximum value of all the pixels in the input

Operation	Rule
	pixel's neighborhood. In a binary image, if any of the pixels is set to the value 1, the output pixel is set to 1.
Erosion	The value of the output pixel is the minimum value of all the pixels in the input pixel's neighborhood. In a binary image, if any of the pixels is set to 0, the output pixel is set to 0.

CONCLUSION

Based on the proposed algorithms, ant colony optimization and genetic algorithm we can give priority to the above mentioned pre processing steps and fitness function can be calculated thereby we can detect the face at any images irrespective of noise, back ground and poor illumination and intensity.

REFERENCES

[1] M. Dorigo, Optimization, learning and natural algorithms (in Italian), Ph.D. Thesis, Dipartimento di Elettronica, Politecnico di Milano, Italy, 1992.

[2] M. Dorigo, L.M. Gambardella, Ant colony system: a cooperative learning approach to the traveling salesman problem, IEEE Trans. Evol. Comput. 1 (1) (1997) 53–66.

[3] M. Dorigo,V. Maniezzo, A. Colorni, Positive feedback as a search strategy,Tech. Report 91-016, Dipartimento di Elettronica, Politecnico di Milano, Italy, 1991.

[4] M. Dorigo, V. Maniezzo, A. Colorni, Ant system: optimization by a colony of cooperating agents, IEEE Trans. Systems, Man, Cybernet.-Part B 26 (1) (1996) 29–41.

[5] M. Dorigo, T. Stützle, Ant Colony Optimization, MIT Press, Cambridge, MA, 2004.

[6] M. Dorigo, M. Zlochin, N. Meuleau, M. Birattari, Updating ACO pheromones using stochastic gradient ascent and cross-entropy methods, in: S. Cagnoni, J. Gottlieb, E.Hart,M. Middendorf, G.R. Raidl (Eds.), Applications of Evolutionary Computing, Proc. EvoWorkshops 2002, Lecture Notes in Computer Science, Vol. 2279, Springer, Berlin, Germany, 2002, pp. 21–30.

[7] L.J. Fogel, A.J. Owens, M.J. Walsh, Artificial Intelligence Through Simulated Evolution, Wiley, New York, 1966.

[8] C. Gagné,W.L. Price, M. Gravel, Comparing an ACO algorithm with other heuristics for the single machine scheduling problem with sequence-dependent setup times, J. Oper. Res. Soc. 53 (2002) 895–906.

[9] L.M. Gambardella, M. Dorigo, Ant colony system hybridized with a new local search for the sequential ordering problem, INFORMS J. Comput. 12 (3) (2000) 237–255.