

Content Based Coding of Face Images

Raman Arora and Hetal Pathak

December 11, 2003

Contents

1	Project Overview	2
2	Introduction	4
2.1	Need for Compression	4
3	Content-based Coding of Face Images using Wavelet Transform	7
3.1	Stage I : Image Segmentation based on Chrominance	7
3.2	Stage II : Detecting Face Regions	8
3.3	Stage III : Face detection based on Energy distribution of Wavelet coefficients	11

List of Figures

2.1	Image compression systems.	6
3.1	Face template for binary template matching.	9
3.2	Binary template matching: (a) a match, (b) no match.	10
3.3	Reorganization of a wavelet coefficients into <i>a wavelet block</i> . The procedure is illustrated for four-level DWT. The DWT coefficients are usually displayed and grouped by subbands, but in this example, coefficients in the same location, but different subbands are grouped together in a block. H,V, and D represents the horizontal, vertical, and diagonal subimages respectively.	12

Chapter 1

Project Overview

The wavelet transform has recently emerged as a promising technique for signal processing applications due to its flexibility in representing the non-stationary signals. The wavelet representation provides a multiresolution/multifrequency expression of a signal with localization in both time and frequency. It decomposes a given signal into a set of coefficients associated with multiscaled wavelets. This property is very much desirable in image and video coding applications, as coding schemes and parameters can be adapted to the statistical properties for each of wavelet coefficients. Wavelet transform based image and video coding techniques have the advantage that they are free from the *blocking artifacts* due to the nature of its global decomposition.

Among all the schemes, content-based image and video coding techniques provide the best image quality at low bit rates. In these techniques, knowledge of the underlying image and video is exploited to achieve the best results. One example of this could be the content-based coding of face images. Face images form the important database in the police departments, banks, security kiosks, and they are found in abundance in day-to-day life. In these databases, the important content of course is the face region. The image coding techniques which masks the face regions for discriminative quantization will be of most importance. These approaches can incorporate human visual system (HVS) aspects into the coding schemes, as the final viewer is going to be human. A promising technique in which HVS can be easily integrated are the wavelet transform based coding schemes.

Video compression at low bit rates has attracted considerable attention recently. This is due to the expanding list of very low bit rate (i.e., < 32 Kbps) coding of “head-and-shoulder” video sequences typical of real-time multimedia, videoconferencing, and videotelephony applications. In “head-and-shoulder” sequences, the human face and the associated subtle facial expressions need to be transmitted and reproduced as faithfully as possible at the receiver. The perceptual quality of such “head-and-shoulder” sequences can be improved by masking the face region for a discriminative quantization to achieve better coding gain.

To use the knowledge of the face regions while coding face images and video se-

quences, the location of the face in a given frame has to be known. To locate the face in a given frame, a wavelet transform based face detection technique has been employed. The main feature used for face detection is the skin-tone color of human beings.

Because of the time-constraint, we were not able to experiment our face detection algorithm with any video codec. The results for the JPEG however are very encouraging. The discriminative quantization of face image keeps the subjective quality of region of interest, i.e. the face, exactly the same while blurring the not-so-interesting background slightly. What we gain out of this is an extra image compression of 40% over the JPEG.

MPEG basically being an extension of JPEG should give similar results. This technique can easily be extended for very low bit-rate (i.e., ≤ 32 Kbps) coding of head-and-shoulder video sequences typical of real-time multimedia applications, video-conferencing, and videotelephony. In head-and-shoulder sequences, the human face and the associated subtle facial expressions need to be transmitted and reproduced as faithfully as possible at the receiver. The perceptual quality of such head-and-shoulder sequences can be improved by masking the face region for a discriminative quantization. The temporal redundancy can further be exploited thereby cutting down computational cost of face-detection. After face has been properly detected, it can be tracked through motion vector and multi-resolution motion estimation (MRME). After every few frames, the face-detection could be carried out again to account for the error accumulated in motion compensation.

Chapter 2

Introduction

Efficient digital representation of image and video signals has been the subject of considerable research over the past 20 years. Modern image and video coding techniques offer the possibility to store or transmit the vast amount of data necessary to represent digital images and video in an efficient and robust way. New audio-video applications in the field of communications, multimedia, and broadcasting became possible based on these digital image and video coding techniques.

2.1 Need for Compression

It is now well agreed that among the necessary ingredients for the widespread deployment of image and video communication services are standards for the coding, compression, representation, and transport of the visual information. These standards, e.g., ITU-T H.261, H.263, H.264, ISO/IEC JPEG, MPEG-1, MPEG-2 and MPEG-4 address a wide range of applications having different requirements in terms of bit rate, picture quality, complexity, error resilience and delay.

Image coding generally involves compressing and coding a wide range of still images, including bilevel or fax images, photographs (continuous tone color or monochrome images), and document images containing text, handwriting, graphics, and photographs. Table 2.1 shows the uncompressed size needed for bilevel (fax), greyscale and color still images.

Unlike speech signals, which can take advantage of a well understood and highly accurate physiological model of signal production, image signals have no such model to rely on. As such, in order to compress and code image signals, it is essential to take advantage of any observable redundancy in the signal. The two most important forms of signal redundancy in image signals are *statistical redundancy* and *subjective redundancy* (also known as irrelevance).

Statistical redundancy takes a variety of different forms in an image, including correlations in the background, correlations across an image, and spatial correlations that occur between nearby pixels.

Table 2.1: Characteristics and uncompressed bit rates for image signals

Image Type	Pixels per Frame	Bits/pixel	Uncompressed size
FAX (200 dpi)	1700 × 2200	1	3.74 Mb
Greyscale	640 × 480	8	2.46 Mb
Color	1024 × 768	24	18.87 Mb

Subjective redundancy takes advantage of the *human visual system* that is used to view the decompressed and decoded images.

Digital image compression maps an original image into a bit stream suitable for communication or storage purposes. The number of bits required to represent the coded image should be significantly smaller than that required for the original image so that one can use less storage space or communication time. There are two basic types of compression, *Lossless* and *Lossy*. *Lossless* compression, which is also called noiseless coding, data compaction, entropy coding, or invertible coding, refers to the algorithms that allow the original pixel intensities to be perfectly recovered from the compressed representation. *Lossy* compression algorithms do not allow that.

A general system for digital image compression is shown in Fig. 2.1(a). It consists of one or more of the following operations, which may be combined with each other or with additional signal processing:

- *Signal decomposition*: The image is decomposed into several images for separate processing, typically by linear transformation, like Fourier or discrete cosine transform or by filtering with a subband or wavelet filter bank. The goal is to concentrate energy in a few coefficients, to reduce correlation, or to provide a useful data structure.
- *Quantization*: High-rate digital pixel intensities are mapped into a relatively small number of symbols. This operation is nonlinear and non-invertible; it is “lossy”. Quantization can include throwing away some of the components of the signal decomposition step. The conversion can operate on individual pixels (scalar quantization (SQ)) or groups of pixels (vector quantization (VQ)).
- *Lossless compression*: Further compression is achieved by lossless coding such as Huffman, or arithmetic coding. The idea here is to assign codewords with a few bits to likely symbols and codewords with more bits to unlikely symbols so that the average number of bits is minimized.

A variety of image compression systems have been proposed, which involve various choices for each of the above three basic components. The JPEG still-image compression standard, for example, uses a discrete cosine transform for the first step, SQ

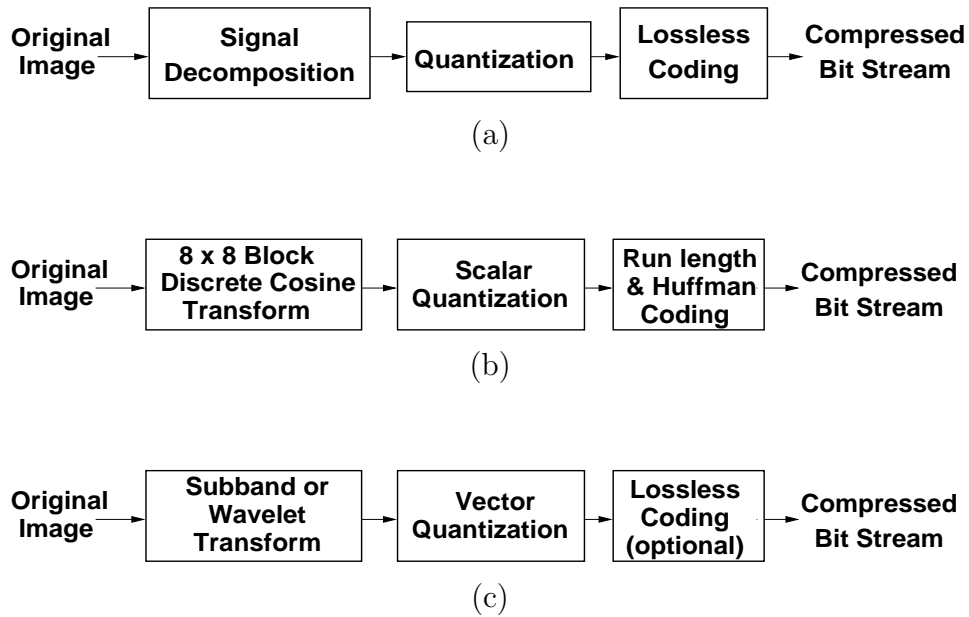


Figure 2.1: Image compression systems.

with different quantizer step sizes for the different transform coefficients in the second step, and run-length coding combined with Huffman coding for the third step, as shown in Fig. 2.1(b). A lossy DCT coder produces characteristic distortion or “artifacts” due to the quantization process. They include “blocking artifacts”, where the block structure used by coder becomes apparent in the decoded image, and “mosquito noise”, where edges in the image are surrounded by fine lines. DCT based image coding systems can provide compression of between 10 and 20 times while maintaining reasonably good image quality. The compression efficiency depends to some extent on the content of the image: an image with lots of details will contain many non-zero high-frequency components and will produce more coded data than a similarly sized image with less detail. Compression can be increased by increasing the quantization scale factor. In general, higher compression can be achieved at the expense of poorer decoded quality.

The image compression systems that use subband or wavelet decomposition for the first step, some form of VQ for the second step, and any lossless coding (or none at all) for the third step, is as shown in Fig. 2.1(c). These techniques have the advantage that they do not require the image to be subdivided into blocks, so the characteristic blocking artifacts of DCT-based systems are avoided.

VQ can be used for many other types of image processing besides compression, and subband/VQ systems can similarly be used for non-compressive purposes as well. For example, wavelet/VQ systems have been used for signature verification, and other clustering and classification uses.

Chapter 3

Content-based Coding of Face Images using Wavelet Transform

In this project We are looking at the class of images for which the region of interest is the face. Such images find lot of utility in day-to-day life. In this section we discuss a fast algorithm that automatically detects faces in wavelet transform domain. Then we can mask our region of interest and discriminately quantize the rest of the image.

The face detection algorithm consists of three stages, where chrominance, shape and the frequency information are used respectively. The algorithm starts at the LL subimage, a lower resolution version of the image obtained from the wavelet transform, so that the amount of data to be processed is greatly reduced. The lower resolution image is sufficient for detection of face regions rather than detailed low-level features. Sophisticated pixel-domain analysis can be applied to the detected regions if needed. This also provides a promising direction for efficient and accurate *face recognition*.

3.1 Stage I : Image Segmentation based on Chrominance

In the first stage of the algorithm, each pixel in the LL subimage corresponding to chrominance components is checked to see if it is a candidate face pixel or not. The key of this classification is the uniqueness of human skin-tone colors.

Human skin tones form a special category of colors, distinctive from the colors of most other natural objects. Although skin colors differ from person to person, they are distributed over a very small area in the chrominance plane (such as (Hue, Saturation or U,V)). The major difference between skin tones is intensity (corresponding to the luminance value Y). In image/video transmission and storage, colors are usually separated into luminance and chrominance components to exploit the fact that human eyes are less sensitive to chrominance variations. We use U and V chrominance components as they are usually used in compression standards, such as, JPEG and MPEG, and

we are also going to use this wavelet transform domain based face detection algorithm for content based compression of color photographs.

The human skin-tone is such that $0.3 \leq \text{Hue} \leq 0.7$ and $\text{Saturation} \leq 0.2$ or in chrominance domain $0.3 \leq Cb \leq 0.5$ and $0.5 \leq Cr \leq 0.7$. After this classification, a binary mask image for a given image is obtained. Each value in the mask indicates the classification results of the corresponding macroblock.

We apply the above classifier to (U, V) values corresponding to LL subimages of chrominance to check for candidate face pixels. As we are using four levels of wavelet transform, each pixel in LL subimage corresponds to 16×16 pixels in the original image. So, whenever any pair of (U, V) gets classified as skin pixel, it means that the corresponding area of 16×16 pixels with respect to this pair is a face block. After the classification, a binary mask image is obtained for each image, but with reduced resolution. Each value in the mask image indicates the classification results of the corresponding block of size 16×16 in the original image. To remove noise and fill in the holes, a 3×3 median filter (which takes into account only first order neighborhood) is applied to the above generated binary mask image.

3.2 Stage II : Detecting Face Regions

Shape Constraints on Human Face

Chrominance information alone is not enough to detect face regions. In an image and video sequences with complex scenes, beside human faces, there may be other exposed parts similar to skin-tones. All these examples will produce positive yet false detections in Stage 1 of the algorithm. In Stage 2, we apply shape constraints of human faces on the binary mask images generated by Stage 1 to eliminate these false alarms and detect candidate face regions.

Just like color, the shape of human face is unique and consistent. The outline of human face can be approximated by an ellipse. As an approximation, the rectangles with certain aspect ratios as the boundary of the face regions is used. The range of aspect ratio of these bounding rectangles is between $[1, 1.4]$. Besides certain aspect ratios, these rectangles are also bounded by size. The size of the image upper bounds the size of face regions. It is lower bounded as well, because it is generally believed, in the face recognition field, that 32×32 pixels is the lower limit for face detection. Since we are working on low resolution LL images, we set the lower limit of our face detection algorithm to 2×2 which corresponds to 32×32 pixels with respect to actual image resolution.

Detecting Face Regions by Binary Template Matching

Stage 1 of the proposed algorithm generates candidate face macroblocks after skin-tone classification. Compared with original images, the resolution of the mask images

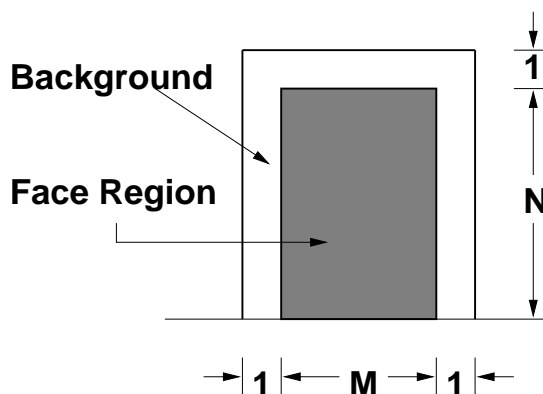


Figure 3.1: Face template for binary template matching.

is 16 times lower in both horizontal and vertical directions. The task is now to find contiguous regions in the mask images that can be bounded by a rectangle satisfying some size and shape constraints. To accomplish this task, binary template matching is used.

The idea is to use rectangles of possible sizes and aspect ratios as face templates to match against the binary mask images. A face template is shown in Fig. 3.1, whose size is $(M + 2) \times (N + 1)$. The actual face template is represented by $M \times N$ pixels.

The template consists of two parts: the face region, which is shaded rectangle of size $M \times N$, and the background, which is the area between the inner and outer rectangles. The size of the face region $M \times N$ should be bounded by size and aspect ratio, according to shape constraints. The reason for considering background as a part of the template is that the color of the background adjacent to the face region is usually distinctive from skin tone, so that there should be few “ones” in this region. The region adjacent to the bottom of the face region is not considered in the template because there can be exposed neck.

The matching criterion is twofold. As this two-frame template slides over the masked image, the number of ones covered in the shaded region and the number of ones covered in the background region are counted. The intuition is that for a match, the first number should be large, and the second one should be small. The number of ones inside the face rectangle, as well as the number of ones in the left, right and top parts of the background region are represented as N_0 , N_1 , N_2 , and N_3 respectively. Only if N_0 is above a threshold, and N_1 , N_2 , N_3 are below certain thresholds, the position of the template is declared as face portion.

This is illustrated in Fig. 3.2. Fig. 3.2(a) is a match, because the face region is almost covered by ones, and there are few ones in the background. Fig. 3.2(b) is not a match, because there are too many ones in the background region.

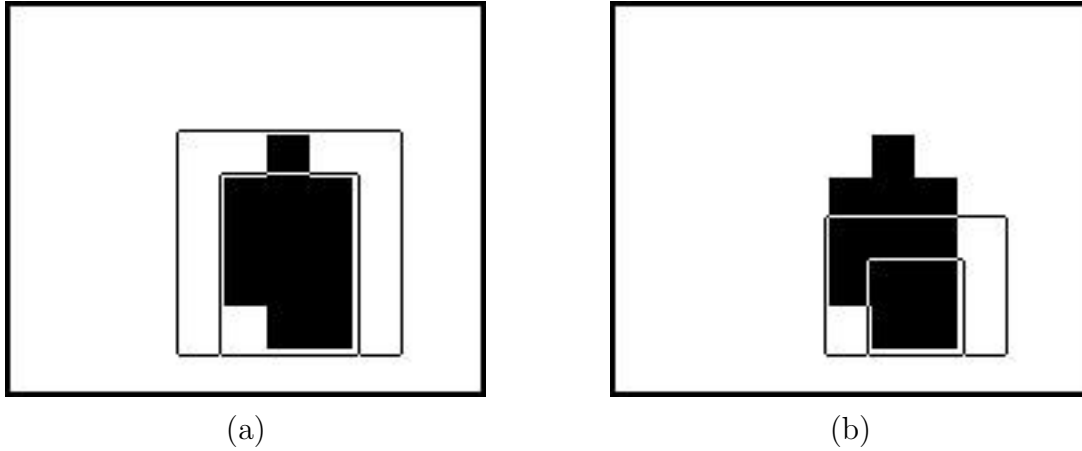


Figure 3.2: Binary template matching: (a) a match, (b) no match.

Segmenting Masked Images

To speed up the processing of binary template matching technique, the search area in the masked image is reduced by segmenting into nonoverlapping rectangular regions that contain either no ones or a contiguous one region. This segmentation is done by projecting the mask image onto the x and y axes. Given a binary mask image $B(x, y)$, ($x = 0, 1, \dots, M_x - 1 : y = 0, 1, \dots, M_y - 1$), the projections are defined as follows, where P_X denotes the projection onto the X axis; P_Y denotes the projection onto the Y axis:

$$P_X(x) = \sum_{y=0}^{M_y-1} B(x, y) \quad x = 0, 1, \dots, M_x - 1$$

$$P_Y(y) = \sum_{x=0}^{M_x-1} B(x, y) \quad y = 0, 1, \dots, M_y - 1.$$

Based on the zero-runs and nonzero-runs in P_X and P_Y , the mask image is segmented into nonoverlapping regions. The way the segmentation is done, it guarantees that in each of the segment, there is either a contiguous one region, or no ones at all.

For those blocks where number of ones is zero, there is no need to perform binary template matching for that segment.

As the size of the face region is unknown, the template matching starts from the smallest possible size, which is 2×2 for masked image corresponding to 32×32 pixels with respect to original, and then gradually increasing the template size in each direction depending on the range of aspect ratio allowed. Therefore, all sizes of the face regions can be detected.

Finally, overlapping face bounding rectangles are resolved. If only a small area is overlapped, this may be a situation where two faces are very close to each other, therefore, both regions are kept as valid face regions. If one of the regions is small and the overlapping area is large compared with its size, the smaller one is discarded and only bigger one is retained as valid face region.

Morphological Processing Approach

An alternative to the template matching is the morphological approach where we not only fill in the holes in the face region detected (corresponding to eyes or moustache) by dilating the detected mask but also remove the false alarms by eroding them with a structure element of size smaller than expected face region.

The morphological operators being used are Dilation and Erosion. Given B is a structuring element, dilation is defined as

$$A \oplus B = \{z | (\hat{B})_z \cap A \neq \phi\} \quad (3.1)$$

Or,

$$A \oplus B = \{z | [(\hat{B})_z \cap A] \in A\} \quad (3.2)$$

Dilation, basically, joins broken edges and fills in holes. Erosion on the other hand removes noise and unwanted details from the image. It is defined as

$$A \ominus B = \{z | (B)_z \subseteq A\} \quad (3.3)$$

So, the binary mask is first dilated with a small structuring element of size 17x17 to fill up the holes in the mask due to eyes or teeth. Then the image is eroded with a structuring element of size 17x17 to restore the image to original shape and size but with holes filled. This is basically closing operation that fills in narrow joints. Finally, the mask is eroded with a structuring element which has size acceptable to face size. This removes false alarms. So the regions that have size less than the size of structuring element are eliminated. The structuring element used is a rectangle of size 4x3. The output of this stage gives the final position of the face by eliminating all the false alarms.

3.3 Stage III : Face detection based on Energy distribution of Wavelet coefficients

The main purpose of the last stage of the algorithm is to verify the face detection result generated by the first two stages and remove false alarms caused by objects with colors similar to skin tones. Because of existence of features, such as, eyes, nose, and lips, there are many discontinuities in intensity level, which gives rise to high frequency wavelet coefficients in luminance component Y. Therefore, using this fact, the false alarms are discarded if enough high frequencies are not present.

Grouping of Wavelet Coefficients

We follow the concept of *wavelet block* proposed for grouping wavelet coefficients in Y based on *zerotree* hypothesis Wavelet coefficients are organized into wavelet blocks as

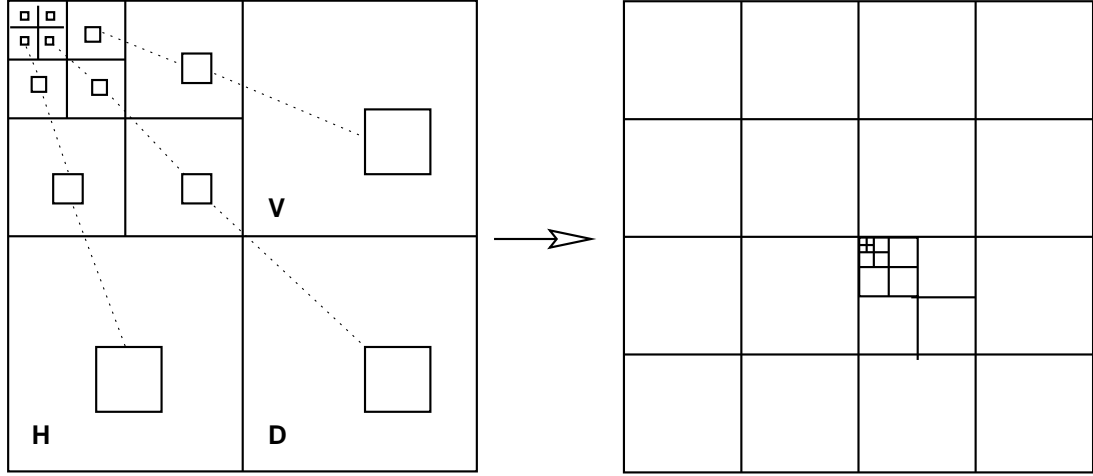


Figure 3.3: Reorganization of a wavelet coefficients into a *wavelet block*. The procedure is illustrated for four-level DWT. The DWT coefficients are usually displayed and grouped by subbands, but in this example, coefficients in the same location, but different subbands are grouped together in a block. H,V, and D represents the horizontal, vertical, and diagonal subimages respectively.

shown in Fig. 3.3, where H, V, and D correspond to horizontal, vertical, and diagonal edge subimages respectively, while upper most left subimage corresponds to coarsest level low pass subimage (L). The concept of wavelet block provides an association between wavelet coefficients and what they represent spatially in the frame.

Verification

Given a candidate face region of size $M \times N$ pixels with respect to masked image, the energy of the corresponding luminance blocks in the DC and H and V areas is calculated as below by pruning the wavelet transform coefficients:

$$E = L(x, y)^2 + \sum_{l=0}^3 \sum_{m=0}^{2^l-1} \sum_{n=0}^{2^l-1} \left[H_{4-l}(m + 2^l x, n + 2^l y)^2 + V_{4-l}(m + 2^l x, n + 2^l y)^2 + D_{4-l}(m + 2^l x, n + 2^l y)^2 \right], \quad (3.4)$$

$$E_{DC} = [L(x, y)^2], \quad (3.5)$$

$$E_H = \sum_{l=0}^3 \sum_{m=0}^{2^l-1} \sum_{n=0}^{2^l-1} [H_{4-l}(m + 2^l x, n + 2^l y)^2], \quad (3.6)$$

$$E_V = \sum_{l=0}^3 \sum_{m=0}^{2^l-1} \sum_{n=0}^{2^l-1} [V_{4-l}(m + 2^l x, n + 2^l y)^2], \quad (3.7)$$

where L corresponds to coarsest level low pass subimage. E , E_{DC} , E_H , and E_V are the total, DC, horizontal, and vertical energies of a single wavelet block. For face region of size $M \times N$, these energies are obtained as follows:

$$E_{MN} = \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} E_{i,j}, \quad (3.8)$$

$$E_{DC_{MN}} = \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} E_{DC_{i,j}}, \quad (3.9)$$

$$E_{H_{MN}} = \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} E_{H_{i,j}}, \quad (3.10)$$

$$E_{V_{MN}} = \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} E_{V_{i,j}}. \quad (3.11)$$

Eqn. 3.8 gives the total energy of all the wavelet blocks in the candidate face region. It equals the energy of the pixel values of this face region, because of the wavelet transforms energy-conserving property. E_{MN} , $E_{DC_{MN}}$, $E_{H_{MN}}$, and $E_{V_{MN}}$ are the energies of all the wavelet coefficients in the candidate region of size $M \times N$ corresponding to average (DC), horizontal and vertical information respectively.

If $E_{DC_{MN}}/E_{MN} < Th_{DC}$, $E_{H_{MN}}/E_{MN} > Th_H$, and $E_{V_{MN}}/E_{MN} > Th_V$, then only the candidate block is declared as face block, where Th_{DC} , Th_H , and Th_V are the thresholds values. The reason is that the face region should not have near 100% energy in DC coefficients. Also, the energy corresponding to horizontal and vertical details should be large enough. Using these threshold, the each candidate face region declared by Stage 2 is verified.

Bibliography

- [1] G. Cote, B. Erol, M. Gallant, and F. Kossentini, "H.263+: Video coding at low bit rates," *IEEE Trans. Circuits System for Video Tech.*, Vol. 8, pp. 849-866, Nov. 1998.
- [2] Y. Qi and B. R. Hunt, "A multiresolution approach to computer verification of handwritten signaturs," *IEEE Trans. Image Proc.*, Vol. 4, pp. 870-874, June 1995.
- [3] S. G. Mallat, "Multiresolution channel decomposition of images and wavelet models," *IEEE Trans. Acoustic, Speech, and Signal Proc.*, Vol. 37(12), pp. 2091-2110, Dec. 1989.
- [4] S. G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Pattern Anal. and Machine Intelligence*, Vol. 11(7), pp. 674-693, July 1989.
- [5] S. G. Mallat, "Mutliresolution approximation and wavelet orthogonal bases of $L^2(R)$," *Trans. Amer. Math. Soc.*, Vol. 315, pp. 69-87, 1989.
- [6] O. Rioul and M. Vetterli, "Wavelets and signal processing," *IEEE Signal Proc. Magazine*, Vol. 8, pp. 14-38, Oct. 1991.
- [7] IEEE Transactions on Information Theory, special issue on wavelet transforms and multiresolution signal analysis, March 1992.
- [8] IEEE Transactions on Signal Processing, special issue on wavelet and signal processing, Dec. 1993.
- [9] Proceedings of IEEE, special issue on wavelet, April 1996.
- [10] Jayashree Karlekar and U. B. Desai, "SPIHT Video Coder," to appear in Proceedings of IEEE Region Ten Conference, pp. 45-48, New Delhi, Jan. 1998.
- [11] Jayashree Karlekar and P. G. Poonacha, "Image Compression using Zerotree of Wavelet coefficients and Fractals," *Proceedings of National Conference on Communication (NCC '97)*, Madras, Feb. 1997.

-
- [12] Jayashree Karlekar, P. G. Poonacha and U. B. Desai, "Image Compression using Zerotree of Wavelet coefficients and Multistage Vector Quantization," *Proceedings of Int. Conf. on Image Proc. (ICIP '97)*, Santa Barbara, USA, Oct. 1997.
 - [13] Jayashree Karlekar and U. B. Desai, "Wavelet based Motion Compensation and Video Compression Scheme," *Proceedings of National Conference on Communication (NCC '99)*, pp. 547-553, Kharagpur, Jan. 1999.
 - [14] Gonzalez, *Digital image processing*,.