

# UC San Diego

## UC San Diego Previously Published Works

### Title

Combining vector quantization and histogram equalization

### Permalink

<https://escholarship.org/uc/item/0672k49r>

### Authors

Cosman, P C  
Riskin, Eve A  
Gray, Robert M

### Publication Date

1991-04-01

Peer reviewed

# Combining Vector Quantization and Histogram Equalization

Pamela C. Cosman  
Information Systems Laboratory  
Stanford University  
Stanford, CA 94305-4055

Eve A. Riskin  
Dept. of Electrical Engineering, FT-10  
University of Washington  
Seattle, WA 98195

Robert M. Gray  
Information Systems Laboratory  
Stanford University  
Stanford, CA 94305-4055

## 1 Introduction

Histogram equalization is a contrast enhancement technique in which each pixel is remapped to an intensity proportional to its rank among surrounding pixels. Histogram equalization is a competitor of interactive intensity windowing, which is the established contrast enhancement technique for medical images. We present work in which histogram equalization is performed on the codebook of a tree-structured vector quantizer. Encoding with the resulting codebook performs both compression and contrast enhancement. It is also possible to perform intensity windowing on the codebook, or a combination of intensity windowing and histogram equalization, so that these need not be done as separate post-processing steps.

## 2 Histogram Equalization

Histogram equalization refers to a set of contrast enhancement techniques which attempt to “spread out” the intensity levels occurring in an image over the full available range [2]. In global histogram equalization, one calculates the intensity histogram for the entire image and then remaps each pixel’s intensity proportional to its rank among all the pixel intensities. In adaptive histogram equalization, the histogram is calculated only for pixels in a context region, usually a square, and the remapping is done for the center pixel in the square. This can be called “pointwise” histogram equalization because, for each point in the image, the histogram for the

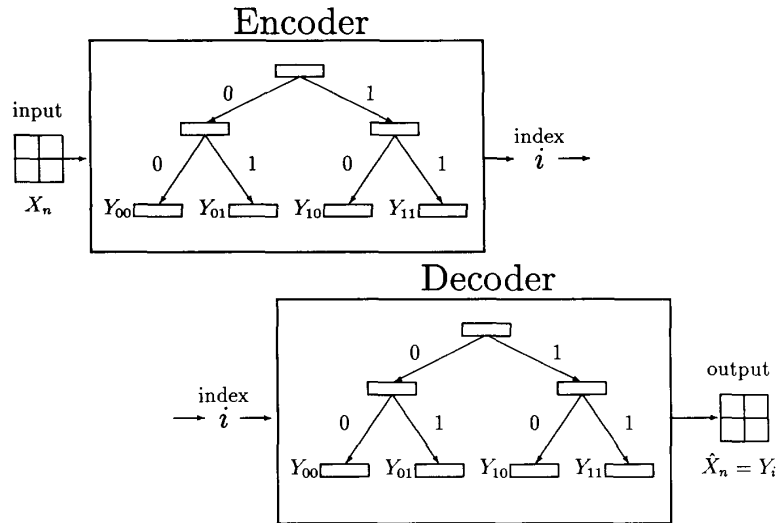


Figure 1: Schema of a vector quantizer

square context region centered on that point is calculated. Because this is too computationally intensive, the bilinear interpolative version calculates the histogram for a set of non-overlapping context regions that cover the image [3]. The remapping of pixel intensity values is then exact for only the small number of pixels that are at the centers of these context regions, and for all other pixels a bilinear interpolation from the nearest context region centers determines the appropriate remapping function.

### 3 Tree-Structured Vector Quantization and Histogram Equalization

Tree-structured vector quantization and histogram equalization can both be applied to one image by performing them sequentially. However, this requires extra time. The tree-structured search by itself is extremely rapid, and a stored compressed image can be retrieved, reconstructed, and displayed in a few seconds on an IBM-RT workstation. Figure 1 shows schematically how this works. The encoder takes the image to be compressed and blocks it into vectors. Each block  $X_n$  is then encoded by a binary tree until it reaches a terminal node (leaf),  $Y_i$ , of the tree, and the index  $i$  of the path through the tree is output. These indices are stored. When the image is to be decompressed, the decoder reads in an index  $i$  and looks at its copy of the tree to retrieve  $Y_i$ , the reconstruction of  $X_n$  [1].

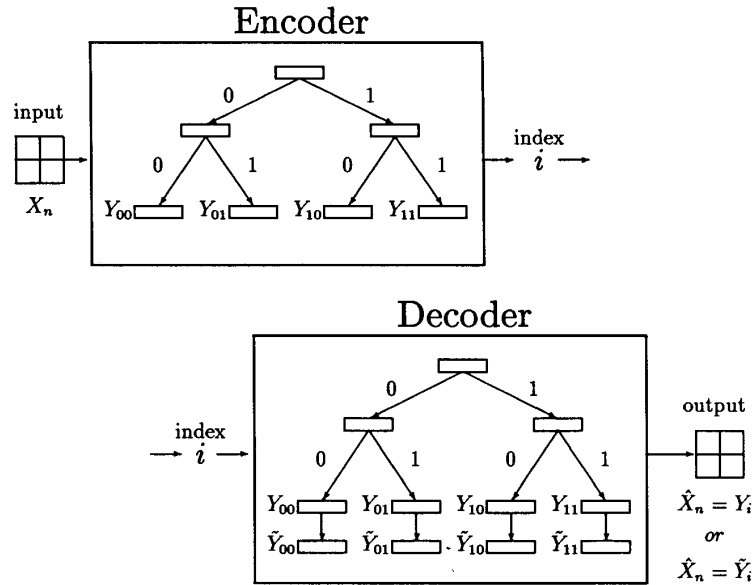


Figure 2: A vector quantizer with both equalized and unequalized codebooks

Adaptive histogram equalization, even the relatively fast interpolative version, takes more than a minute on an IBM-RT workstation for a 512-by-512 image, and this is unacceptable for a radiologist wishing to view, for example, a 20-image study. Global equalization is much faster, but still can take longer than a decoding operation. Instead of performing the operations sequentially, one can perform them simultaneously by equalizing the decoder's codebook off-line, so that the decoder's reconstruction of the image and the equalization would be performed in the same time required by the decompression alone. For example, one can construct a global histogram containing all pixels that composed the training images, and each pixel of each codeword can be equalized using this global histogram. Thus, each pixel of the codebook will be remapped to a new intensity that is proportional to its rank in the global histogram. These new codewords can be stored at the decoder, along with the original codewords. The new system is diagrammed in Figure 2. The encoder is unchanged. The decoder takes the same set of indices and puts them through the same tree, but, upon reaching a terminal node of the tree, the decoder now has the option of outputting either  $Y_i$  or  $\tilde{Y}_i$ . The radiologist thus has the option of looking at either the equalized or the unequalized series of scans and either way requires the same amount of time to reconstruct the image.

To demonstrate the new technique of combining the compression and histogram equalization steps, a tree was grown to a depth of 2 bits per pixel (bpp) on a training sequence of 5 magnetic resonance (MR) mid-sagittal brain scans of 5 different subjects. The training images were blocked into 2 by 2 vectors. The tree was pruned back to 1.7 bpp and used to encode a test image not in the training set. Figure 3 shows the original test image. The histogram equalized version of the compressed image is reconstructed by inputting the same set of codeword indices to a tree whose terminal node codewords were equalized over the global histogram produced by all 5 training images (with the background removed). Figure 4 shows the regular and equalized compressed images.

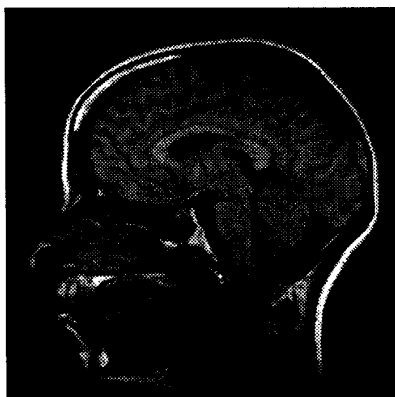


Figure 3: Original image

## 4 Intensity Windowing

Interactive intensity windowing is the established contrast enhancement technique for medical images. It is routinely used for some modalities, such as CT, in which the original images have a dynamic range of 12 bits and cannot be displayed properly on 8-bit monitors. It consists of a thresholding operation followed by a linear rescaling of intensity values. For example, the intensities over 500 can be mapped to 500, and then the range of 0 to 500 can be linearly rescaled to 0 to 255, for an 8-bit display. This windowing can be done automatically by the codebook. Each pixel in each codeword can be thresholded at 500 and linearly rescaled off-line. Use of that codebook for image decompression would then include the intensity windowing step. As an improvement over linear rescaling, the thresholded codewords can be globally histogram equalized first and then rescaled.



Figure 4: Compressed images: regular and equalized

## 5 Future Work

Global histogram equalization, while providing some improvement in image quality as shown in Figure 4, does not provide as much detail in the resulting image as does adaptive histogram equalization (AHE). We are attempting to combine AHE with image compression. When the training images are blocked into vectors, the original  $x,y$  coordinates of the vectors can be saved in addition to the intensity information. When the tree is completed, there will be a record of where in the original training images the vectors that mapped into each leaf of the tree originated. The pixels in that leaf can then be equalized using context regions centered in that location in each training image. The tree could also be constrained in the growing process to cluster training vectors together based on their spatial nearness as well as their similarity in intensity.

Another way to combine AHE and compression is to code the training images using the codebook and also to histogram equalize them adaptively. Each vector in training images is marked by the codeword to which it is closest. Then, each codeword in the AHE codebook can be formed simply by clustering the associated AHE vector of all the training vectors that mapped into that codeword. The motivation is that on average, training vectors from common regions of the image will be assigned to like codewords.

## 6 Conclusion

We have shown that global histogram equalization using the histogram of the training images can be performed off-line on a vector quantizer's codebook. When

an encoded image is decoded in real time using this equalized tree, the resulting decoded picture is very similar to what would be produced by decoding on the regular tree, followed by a global equalization of the decoded image using its own histogram. The time required for this post-processing step is no longer needed. Intensity windowing, or some combination of intensity windowing and histogram equalization, can also be done off-line on the codebook.

## References

- [1] A. Buzo, A. H. Gray Jr., R. M. Gray, and J. D. Markel. Speech coding based upon vector quantization. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 28:562–574, October 1980.
- [2] R. Hummel. Image enhancement by histogram modification. *Computer Graphics and Image Processing*, 6(2):184–195, April 1977.
- [3] S. Pizer, E. Amburn, J. Austin, R. Cromartie, A. Gesselowitz, B. ter Haar Romeny, J. Zimmerman, and K. Zuiderveld. Adaptive histogram equalization and its variations. *Computer Vision, Graphics, and Image Processing*, 39(3):355–368, September 1987.