Computer Assignment 4: Self Organized Feature Maps and Image Data Compression

Christopher Robbiano — Due: May 4, 2018 — ECE656

1 Introduction

The purpose of this computer assignment is to design a self-organized feature map (SOFM) to perform vector quantization (VQ) on digital images. The aim is to perform data compression for image storage or transmission by reducing the number of bits for efficient representation of images. Several widely used images are used in this assignment, namely Lena, Baboon, Peppers and the Boat image. These images are of size 512x512 pixels with 8 bits per pixel (bpp) resolution.

2 Theory

SOFM aim to represent the natural topological organization of neurons that exist in the human brain, implementing the Hebbian learning rule motto of 'neurons that fire together, wire together'. Biological neural networks form of 2-D layers of neurons densely interconnected by lateral feedback synapses. The strength of these connections vary as a function of lateral distance. A neighborhood of 50-100 micron radius is characterized by short-range excitatory connections while 200-500 micron rings are longer-range (weaker) inhibitory connections. The feedback connection weights are shaped like Mexican hat. The self-feedback is positive. An approximation for lateral connection function used in SOFM is

$$y_i(k+1) = f\big(x_i(k+1) + \sum_{m=-m_0}^{m_0} \gamma_m y_{i+m}(k)\big)$$

where $x_i(k)$ and $y_i(k)$ are the input and output neurons i at time k and γ_m is the feedback coefficient for the sampled Mexican hat. The second term show the effect of all the lateral connections within the excitatory region of $m \in [-m_0, m_0]$.

At learning step k and given neighborhood N_m around the winning cell m, all cells within N_m are updated, while those outside of the neighborhood are left unchanged. The winner cell m is selected using $||\mathbf{x} - \mathbf{w}_m|| = \min_i ||\mathbf{x} - \mathbf{w}_i||, \forall\ i \in N_m$. This implies that SOFM requires both competition and cooperation between neurons at the same time. The radius of N_m varies with time throughout the learning process until the neighborhood encapsulates only the winning cell.

The update equations are given by

$$\begin{aligned} \mathbf{w}_i(k+1) &= \mathbf{w}_i(k) + \alpha(k) h_{i,m}(k)[\mathbf{x}_k - \mathbf{w}_i(k)] & \qquad i \in N_m(k) \\ \mathbf{w}_j(k+1) &= \mathbf{w}_j & \qquad j \notin N_m(k) \end{aligned}$$

where $0 < \alpha(k) < 1$ and $h_{i,m}(k)$ must be symmetric and a decreasing function of lateral distance $d_{i,m}$ between the winning neuron m and other neurons $i \in N_m$. Typically a Gaussian function is used to satisfy these requirements. The learning moves the weight vector of the winning neuron closer to the input vector $\mathbf{x}_k$ while the rest of the neurons only get a fraction of the correction.

3 Data Processing, Training and General Procedure

All of the data processing was done with Matlab. The 4 images used in this assignment are seen in the bottom rows of each sub image in Figure 3 below. For each image in the image set the following steps were followed:

1. Segment the original image I to be used for training the SOFM network into 4×4 blocks and then vectorized those blocks and stored them into a large matrix

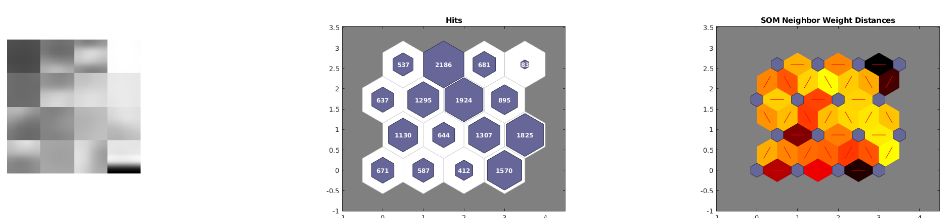
2. Segmented all other images into 4×4 blocks and then vectorized those blocks and stored them into a large matrix

3. Built a SOFM network $\forall\ n \in [16, 32, 64, 128, 256]$ neurons relating to n features for vector quantization

4. Trained the SOFM network until convergence

5. Reconstructed *all* images with the SOFM features built from image I for analysis

At the end of the training we had the weights corresponding to the centers of each quantized feature vector. The features for the 16 feature SOFM are shown in Figure 1a and have been scaled up for your viewing pleasure. An example of one of the SOFM feature maps is shown in Figure 1b, where each hexagonal node (neurons) contains a number representing the number of 4×4 blocks from the original image that map to that neuron. The distances between each neuron in the feature space can be seen in Figure 1c. It can be observed that there are distinct regions within the distance map, i.e. there are neighborhoods that have a smaller distance (darker color) between neurons with a large distance between neighborhoods. This would indicate that there are some fundamentally distinct features that exist within each image.



(a) Extracted 4×4 features (b) # of blocks per feature (c) Distance between features

4 Results and Discussion

The compression rate is a function of block size and the number of features that are chosen to represent the original image. It can be described as $CR = (B \times M \times N)/\big(\log_2(F) \times (N \times M)/(k \times k)\big)$ where B is the number of bits per pixel, M and N are the original image dimensions, F is the number of features to use and k is the block size. The distortion rates were calculated from the averages of all the reconstructed images that were used to generate the SOFM features, i.e. distortion was calculated from only the reconstructed images that generated the features originally, and the values are shown in the performance plot in Figure

2. In the region shown in the figure, the relationship is described best by a linear fit, but it is necessarily so that that a general increasing monotonic trend best describes the trade-off in performance.

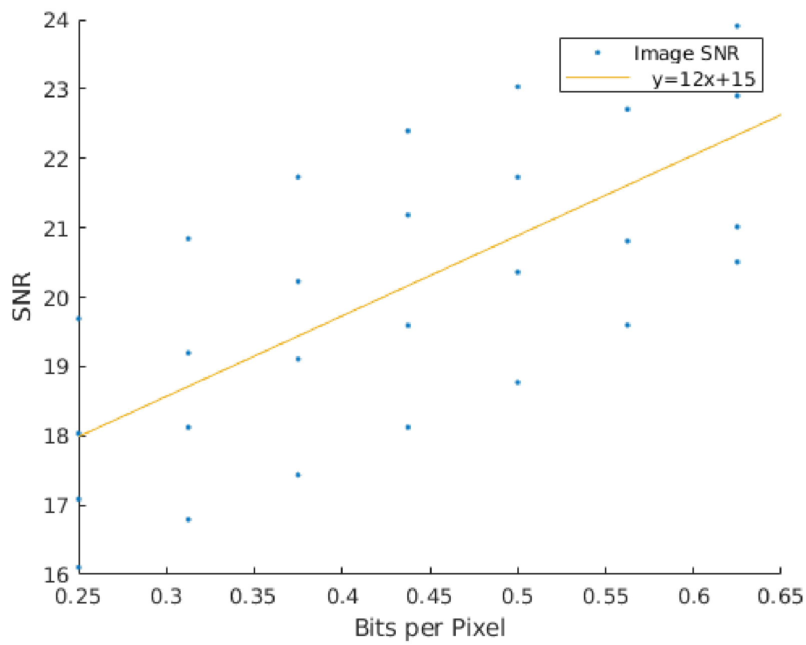


Figure 2: Performance plot of SNR versus bits per pixel

Reconstructed images, using the baboon image as the feature generating image, for each number of features can be seen in Figures 3. The SNR doesn't capture an intuitive feeling for how similar the original and reconstructed images are, and would sometimes report a lower SNR for a objectively better reconstruction. For this reason, the structure similarity index for images (SSIM) was also used to quantify how similar the reconstructed image is to the original image. The SSIM is annotated under each of the reconstructed images and captures the intuitive feeling that a reconstructed image that better resembles the original has a higher similarity index. It is evident that as the number of features increases that so does the visual quality of all images. For reconstruction of the original generating image, 128 feature vectors seems to visually work well, but for all non-generating images it would appear that you need > 512 feature vectors to really reproduce the image without some visual distortion. In other words, you need a large number of feature vectors in order to generalize well.

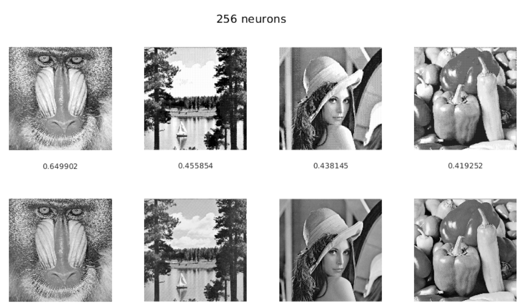
The ability to perform data compression on an individual image is rather impressive, given that you have the codebook associated with that image. For general vector quantization of images without having the codebook generated for the exact image, the data compression abilities remain the same, but the reconstructed image suffers. In this manner, the generalization abilities of the SOFM are moderate at best, especially when comparing

(a) Reconstructions with 16 features

(b) Reconstructions with 32 features

(c) Reconstructions with 164 features

(d) Reconstructions with 128 features



(e) Reconstructions with 256 features

(f) Reconstructions with 512 features

(g) Reconstructions with 1024 features

Figure 3: Original and reconstructed images for varying number of features

this to the performance say the DCT or wavelet decomposition of an image for data compression and reconstruction, the SOFM really tends to pale in comparison.

One very interesting application of SOFM could be generating a mosaic of an image given a particular set of other images. The set of images would act as the feature vectors, and the reconstructed image would be composed of small blocks that were from the set of images.

5 Conclusion

In this assignment we looked at the data compression and vector quantization abilities of SOFM. The performance of the SOFM for compression and reconstruction was evaluated and it was determined that if the codebook containing the feature vectors for a particular image is known then you need only 128 vectors for a convincing reconstruction. This leads to a 18.2857 compression rate and 0.4375 bits per pixel instead of the traditional 8 bits per pixel. In general, the SOFM provides some insight to the topological ordering of the features that compose the original data but performs poorly in signal reconstruction when compared to a DCT basis

6 References

Neural Networks and Learning Machines
ECE656 Class Notes