Neural Network Programming Assignment 1:

Image Decryption by a Single-Layer Neural Network

Professor CCC developed a method of image encryption. This method uses two image keys, say K_1 and K_2 for encrypting an input image, say I. The encryption derives an encrypted image, say E, by mixing the two image keys and the input image. The mixing model is

$$E = w_1K_1 + w_2K_2 + w_3I.$$

Note that these K_1, K_2, and E have the same dimension. To give you an example, Figure 1 and Figure 2 are respectively the two image, K_1 and K_2, keys used in the encryption. Figure 3 is an example image (I) to be encrypted. The encrypted image E derived after the encryption is shown in Figure 4.

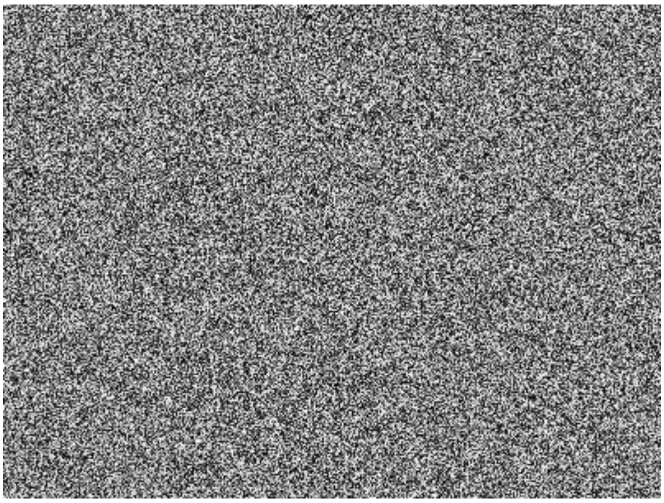


Figure 1. Image Key 1 (K_1).

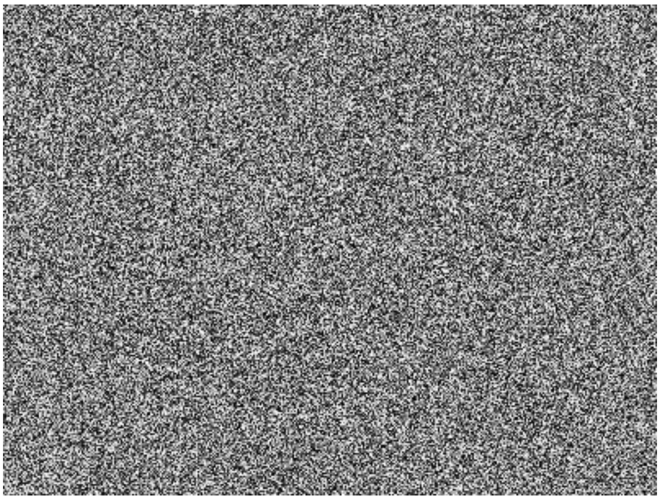


Figure 2. Image Key 2 (K_2).

Figure 3. The input image to be encrypted (I).

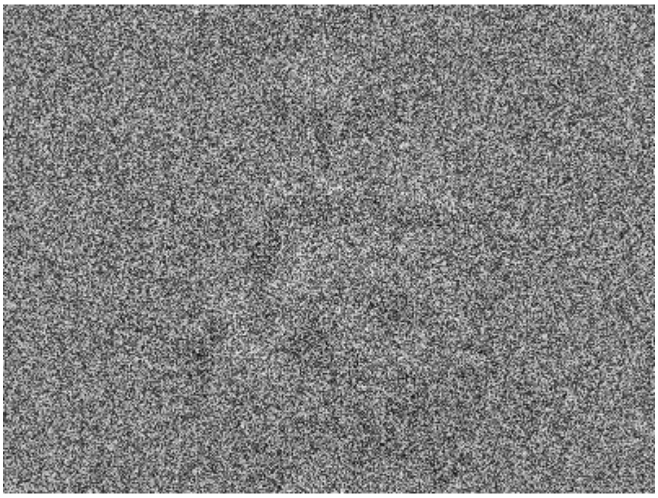
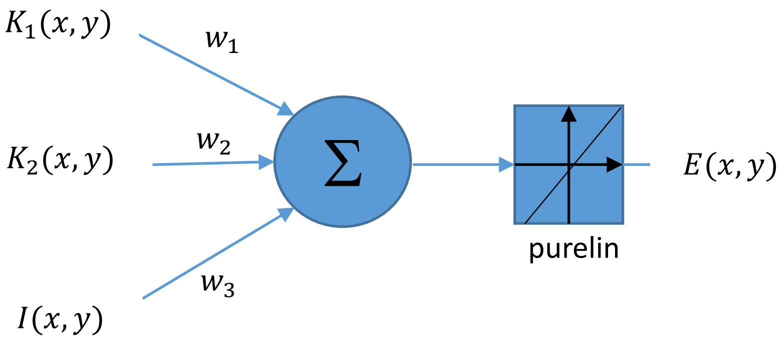


Figure 4. The encrypted image (E).

To decrypt the image I from E, you will need the two image keys and try to find the mixing parameters w_1, w_2, and w_3. Once these requisites are available, you then can derive the original input image by

$$I = \frac{E - \mathrm{w}_1 K_1 - w_2 K_2}{w_3}. \tag{1}$$

Now, you are required to derive the mixing parameters by using a single-layer neural network. The network architecture would be like the following:



By collecting every corresponding pixel k on the four images, K_1, K_2, I (inputs) and E (target outputs), we obtain a training set $\{[K_1(k), K_2(k), I(k)], E(k)\}$ for $1 \le k \le W \times H$. Using this training set, we can learn the mixing parameters $\mathbf{w} = [w_1, w_2, w_3]^{\mathrm{T}}$ by using the training algorithm of the single-layer neural network based on the gradient descent method. That is,

$$\boldsymbol{w}(k+1) = \mathbf{w}(k) + \alpha\big(t(k) - a(k)\big)\mathbf{x}(k),$$

where $\mathbf{w}(k) = [w_1(k), w_2(k), w_3(k)]^{\mathrm{T}}$, $\mathbf{x}(k) = [K_1(k), K_2(k), I(k)]^{\mathrm{T}}$, $a(k) = \mathbf{w}(k)^T\mathbf{x}(\mathrm{k})$, $t(k) = E(k)$, and α is the learning rate. The training algorithm is listed as follows

Algorithm: Adaptive Least Mean Square Error Method by Gradient Descent

Input: training samples $\{[K_1(i), K_2(i), I(i)], E(i)\}_{i=1}^{W\times H}$, where W and H are the width and height of the images, respectively,

MaxIterLimit: the maximal number of epochs allowed to train the network,

ϵ**:** the vigilance level for checking the convergence of weight vectors,

α: the learning rate (suggest to be a small constant such as 0.00001)

Output: $\mathbf{w} = [w_1, w_2, w_3]$

Steps:

1. Set $Epoch$ = 1.
2. Randomize $\mathbf{w}^{\boldsymbol{Epoch}}(0) = \left[w_1^{Epoch}(0), w_2^{Epoch}(0), w_3^{Epoch}(0)\right]^T$
3. While ($Epoch$=1 || $Epoch$<*MaxIterLimit* && $\left\|\mathbf{w}^{\boldsymbol{Epoch}} - \mathbf{w}^{Epoch-1}\right\| > \epsilon$) do
   (1) For k = 1 to $W \times H$
      i. $a(k) = \mathbf{w}^{Epoch}(k)^T\mathbf{x}(\mathrm{k})$, where $\mathbf{x}(k) = [K_1(k), K_2(k), I(k)]^{\mathrm{T}}$
      ii. $e(k) = E(k) - a(k)$
      iii. $\boldsymbol{w}^{\boldsymbol{Epoch}}(k+1) = \mathbf{w}^{\boldsymbol{Epoch}}(k) + \alpha \cdot e(k) \cdot \mathbf{x}(k)$
   (2) End For
   (3) $Epoch$ = $Epoch$ + 1
4. End While

After deriving the mixing parameter $\mathbf{w}$ from the algorithm, you have to decrypt another encrypted image E' given in Figure 5 using K_1, K_2, and the derived $\mathbf{w}$ according to Equation (1). Are you wondering what the original image looks like? If yes, then design the neural network for the decryption right away. You'll find it's a great and interesting accomplishment to retrieve a clean image from such a cluttered image using a simple neural network.

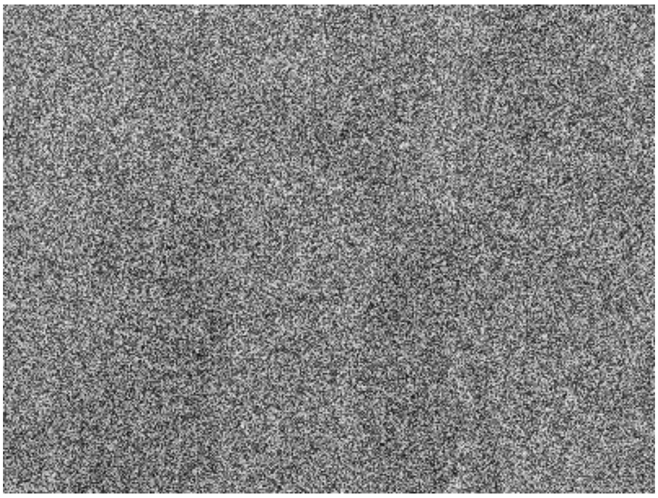


Figure 5. Another encrypted image (E') to be decrypted.

Things you need to submit to the course website include

(1) Source codes with good comments on statements;

(2) A 3~5-page report with

   A. the way how you prepare the training samples

   B. all parameters, such as *MaxIterLimit*, α, and ϵ, you used for the training algorithm,

   C. the derived weight vector $\mathbf{w}$,

   D. the printed image I' decrypted from E',

   E. the problems you encountered, and

   F. what you have learned from this work.

Note: the link to download all required images are given on the course website. The downloaded file is a compressed archive containing five gray images (K1, K2, E, I, and E') in PNG format and five text files with each storing the pixel data of each image. You can choose either the png or the txt for your use. MATLAB is recommended to do this assignment for its powerful capability in handling various kinds of image data.