MATH 179/279 HOMEWORK 4

Due: Tuesday, May 7, 2019.

(1) $\boxed{\textbf{SOURCE CODE:}}$

Implement a Haar wavelet transform and its inverse. Recall that given a function of the form,

$$f_j(x) = \sum_k a_k^j \phi(2^j x - k) \in V_j$$

can be decomposed as

$$f_j(x) = f_{j-1}(x) + w_{j-1}(x),$$

where,

$$f_{j-1}(x) = \sum_k a_k^{j-1} \phi(2^{j-1}x - k) \in V_{j-1},$$
$$w_{j-1}(x) = \sum_k b_k^{j-1} \psi(2^{j-1}x - k) \in W_{j-1},$$

with coefficients,

$$a_k^{j-1} = \frac{1}{2}(a_{2k}^j + a_{2k+1}^j),$$
$$b_k^{j-1} = \frac{1}{2}(a_{2k}^j - a_{2k+1}^j).$$

This gives the decomposition $V_j = V_{j-1} \oplus W_{j-1}$. In practice, since the number of coefficients in the two representations is the same, it is natural to store this data in a vector of length 2^j, where the first 2^{j-1} entries contain the a_k^{j-1}, and the next 2^{j-1} entries contain the b_k^{j-1}. By repeatedly applying this at each level, we eventually represent V_j as $V_0 \oplus W_0 \oplus W_1 \oplus \cdots \oplus W_{j-1}$.

Given an initial vector v of length 2^j, we can consider the coefficients at the j-th level to be given by,

$$a_k^j = v_k.$$

To construct the inverse Haar wavelet transform, observe that,

$$\tilde{f}_{j+1}(x) = \sum_k \tilde{a}_k^j \phi(2^j x - k) + \sum_k \tilde{b}_k^j \psi(2^j x - k) \in V_j \oplus W_j$$

can be written as,

$$\tilde{f}_{j+1}(x) = \sum_k \tilde{a}_k^{j+1} \phi(2^{j+1} x - k) \in V_{j+1},$$

where,

$$\begin{aligned}\tilde{a}^{j+1}_{2k} &= \tilde{a}^j_k + \tilde{b}^j_k,\\ \tilde{a}^{j+1}_{2k+1} &= \tilde{a}^j_k - \tilde{b}^j_k.\end{aligned}$$

By applying this repeated, we eventually recover V_j from the representation in $V_0 \oplus W_0 \oplus W_1 \oplus \cdots \oplus W_{j-1}$.

(2) **APPLICATION PROBLEM:**

We will apply the Haar wavelet transform to the filtering of images.

Use the command,

```
>> A=imread('filename.ext')
```

to load a graphics file into the three-index array `A`, where the first two indices are the x and y position, and the third index is the color channel.

Crop the image so that the x and y directions have lengths that are powers of 2. Apply your Haar wavelet transform to the image, first in the x direction, and then in the y direction, for each of the color channels.

Once you have performed the Haar transform, we will now attempt to discard the least significant entries. To determine the entry of the three-index array `A` with the largest magnitude, use the following,

```
>> max(max(max(abs(A))))
```

You can then set the entries which are below a threshold to zero using,

```
>> B=A; B(abs(A)<threshold)=0
```

You can try different thresholds that are a fraction of the maximum magnitude. Once you have done this, perform the inverse Haar transform to recover an image. The image can be written to disk by using,

```
>> imwrite(A, 'filename.ext')
```