Project 2A: Compression of Audio Signals using Haar Wavelets

Course: MCIT 515 - Fundamentals of Linear Algebra and Optimization

Sorawit Roongruengratanakul; Sunita Pathak

(Problem 3)

For k = 4, The transformed vector starts to show repetitive or symmetric structures. As k increases, the transformation progressively smooths out the signal. Some large flat sections were observed.

(Problem 5.1)

We start by loading and listening to a Handel piece

```
load handel; % Load the built-in Handel audio sample
handel = y(1:65536); % Keep the first 65536 elements
sound(handel, Fs); % Play the original audio
```

```
k = 1;
handel_transformed = haar_step(handel, k);
sound(handel_transformed, Fs); % Play the transformed sound
```

– Song becomes quite tinny

```
k = 2;
handel_transformed = haar_step(handel, k);
sound(handel_transformed, Fs); % Play sound after k=2
```

– Song becomes grainy and muffled, some melody lines sounded missing

---

```
k = 3;
handel_transformed = haar_step(handel, k);
sound(handel_transformed, Fs); % Play sound after k=3
```

– Even grainer, some highs and lows are missing. The midsection is still discernible. Sounds a bit distorted.

Presumably, as the value of k increases, only the very basic structure of this song will remain, trading the highs, lows, and other details in favor of smaller file size due to more compression.

(Problem 5.2)

To analyze compression, we set the second half of the Haar coefficients to zero:

```
c = haar(handel);
c1 = c;
c1(32768:end) = 0; % Zeroing detail coefficients
handel1 = haar_inv(c1);
sound(handel1);
```

The reconstructed audio sounded much more like the original, especially the vocal chorus which sounds almost exactly the same. However, there were some ornamental sounds and instruments that seemed to be missing. This matches the expectations of lacking the finer details when reconstructing audio, but the primary structure is still retained.

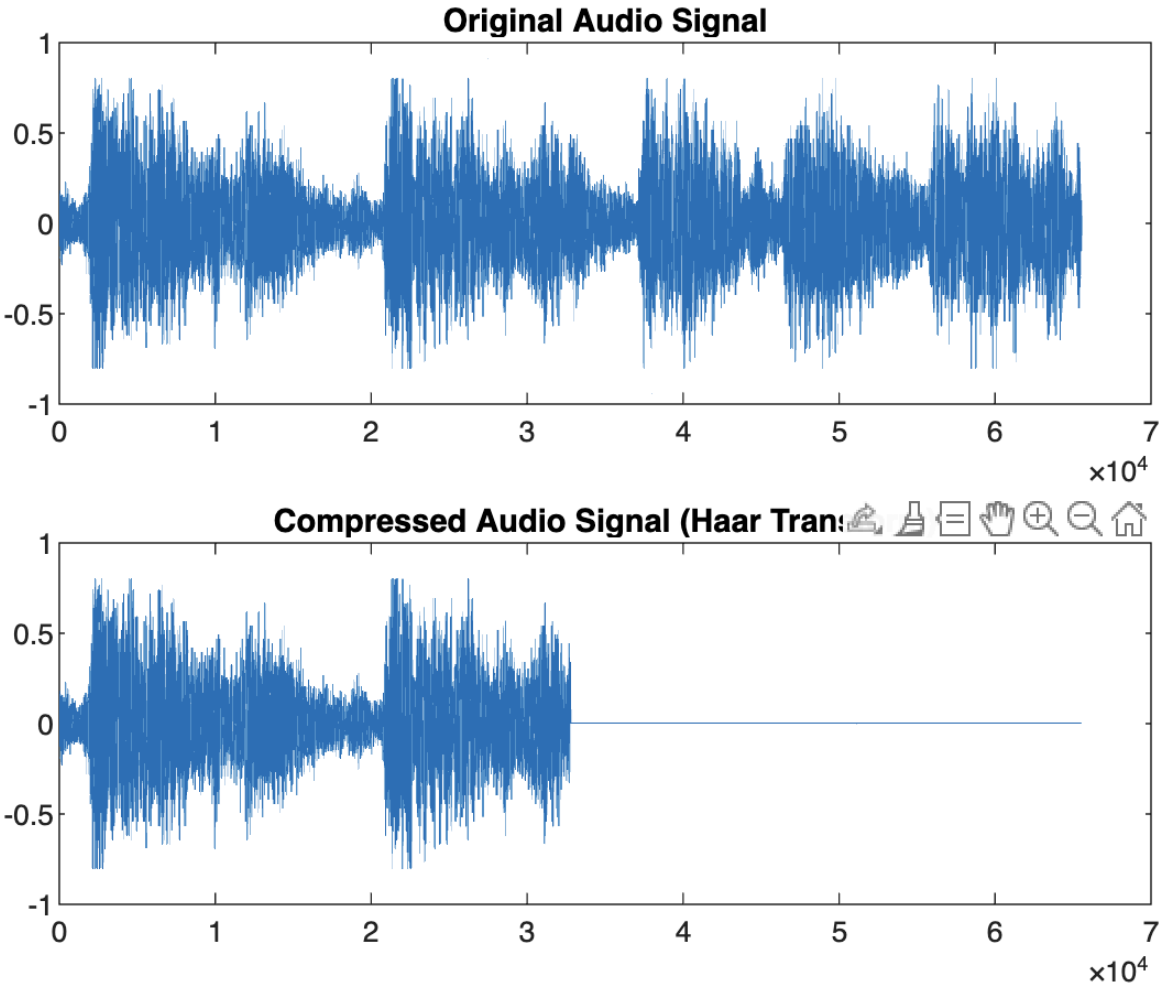
After running

```
c = haar(handel);
c1 = c;
c1(32768:end) = 0;
handel1 = haar_inv(c1);
sound(handel1)
```

The music duration was cut in half. We also graphed this by using

```
figure;
subplot(2,1,1);
plot(handel);
title('Original Audio Signal');

subplot(2,1,2);
plot(handel1);
title('Compressed Audio Signal (Haar Transform)');
```



Which matches up with what we were hearing. Trying again with c1(16384:end) also did the expected thing of quartering the song duration.