

# Retrieving Short and Dynamic Biomedical Sequences

Markus Nilsson

Mälardalen University

Department of Computer Science and Electronics

Högskoleplan 1

P.O. Box 883, SE-721 23 Västerås, Sweden

markus.nilsson@mdh.se

## Abstract

In this paper, we present a method with a low complexity for analysing short and dynamic biomedical sequences. The method uses the Daubechies D4 wavelet in combination with similarity fitness schemes for retrieval. The method has been shown to outperform Fourier based methods in retrieving biomedical sequences of dynamic lengths, as well as the Haar wavelet.

## Introduction

It is not always desirable to analyse entire sessions of time series. There are domains where studying subsets of samples from a longer sequence give more information than studying the entire sequence. Classification of biomedical sequences, for instance RSA (Nilsson & Funk 2004), is such a domain. This specific domain has to handle temporal attributes within the sample sequences. These attributes may appear arbitrarily, as they are physiological measurements. These sequences are also dynamic in length.

Discrete Fourier Transformations (DFT) (Smith 1999) is often the first choice for the analysis of time series. They are well understood and widely used; however, there are limitations when using DFTs. The DFT is not able to detect arbitrary occurring temporal attributes, such as small deviations within a signal in the time domain. The DFT usually requires high number of transformation points in order to not miss any peak frequencies even though the number of samples in the sequence might be very sparse. A common implementation of a DFT is by using a 1024 point FFT (Sterns 2003). Thus, computing large DFTs on a low number of samples might be resource inefficient.

There are methods that solve the issue of a large number of complex Fourier coefficients, as well as assigning generic weights, like the recent D-HS<sup>T</sup> (Patterson, Galushka, & Rooney 2004) method. The D-HS<sup>T</sup> method assumes that lower frequencies contain more information on how to reconstruct the signal than higher frequencies. By using this method it is possible to ensure that there will be no

false dismissals due to Parsevals theorem (Oppenheimer & Schafer 1975), i.e., the Euclidian distance must be smaller if frequencies are removed. This method dramatically reduces the number of Fourier coefficients.

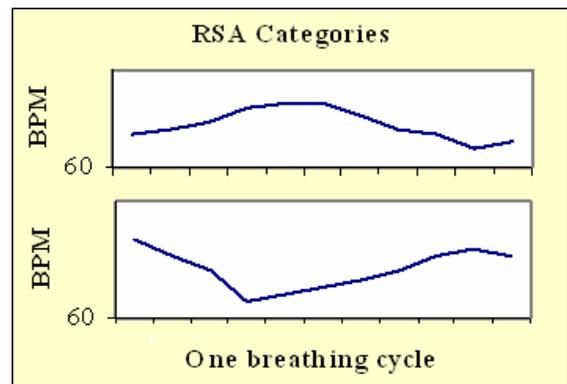


Figure 1: These two sequences look different; they do also belong to different classes. The top graph depicts a normal RSA, the lower has been shifted in time, thus is a dysfunction (von Schéele 1999). A traditional DFT can not tell these two apart since they have identical frequency spectrums.

A limitation with the DFT is its inability to point out when a specific frequency occur, only that it occurs, somewhere in the investigated series of samples. Consider the example in figure 1. A traditional way of solving this issue is by using a Short Time Fourier Transformations (STFT) (Daubechies 1990). The STFT overcomes the time limitation by using a smaller "window" on the samples, i.e., using a subset of samples. The window slides through all samples and calculates a new DFT for each window. This can be visualised by arranging a number of DFTs to form a three dimensional map, the new third dimension, will represent the time. A drawback with the STFT is its static window. A high frequency may oscillate several times within the window if the window is too wide, and a lower frequency may never be detected if the window is too narrow. A dynamic window is needed. We choose to explore wavelets as the dynamic window in this paper. The next section addresses wavelets

together with a method for retrieval, followed by a section of tests and results. The final section in the paper is the conclusions.

### Using Wavelets for Retrieval

Discrete Wavelet Transformations (DWT) (Daubechies 1990; Hippenstiel 2002) breaks down discrete signals (read set of samples) to their principal frequencies based on a function. DWTs use a dynamic function (window) when computing frequencies. This function changes shape depending on what frequencies are to be analysed. A general concept with DWT functions is that they are more accurate in time when analysing higher frequencies and more accurate in frequency interval when analysing lower frequencies. There is always a trade-off between these two parameters. A DWT function returns frequency intervals instead of specific frequencies. This is illustrated in figure 2. The figure illustrates a general template for DWTs. The function returns frequency coefficients in a 2-dimensional scale- and time shift matrix, instead of frequency coefficients like in DFTs. Scale is the frequency band and time shift is when in time the frequencies occur (Bentley & McDonnell 1994). As we can see, the lowest window, i.e.  $C_7$ , analyses the entire length in time but only  $\frac{1}{4}$  of the frequency compared to  $C_1$ .

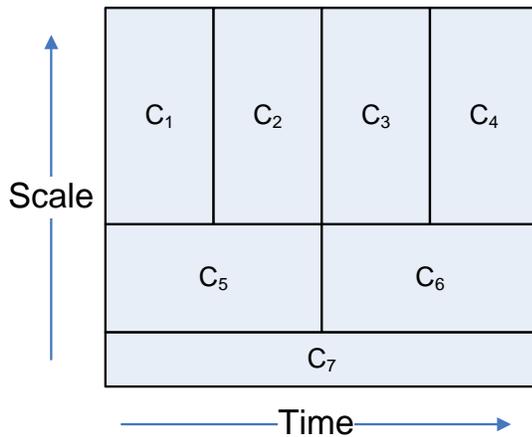


Figure 2: A DWT with 7 frequency coefficients, 4 in the high-band ( $C_1$ - $C_4$ ),  $C_5$  and  $C_6$  represents the mid-band, and  $C_7$  is in the lowest frequency band.

In practical terms, a wavelet function does basically mean average samples in an iterative loop. The DWT function calculates two different variable types, namely frequency coefficients and intermediate scale variables. The scale variables contain smoothed values, based on averages of nearby samples; and the frequency coefficients are the changes in the averaged samples. The coefficients calculated in the same iteration are part of the same frequency band. For  $n$  samples, there are  $\frac{n}{2}$  frequency coefficients and  $\frac{n}{2}$  scale variables. The scale variables act as the input to the next iteration of the DWT function.

Thus, the iterations required are in the order of  $\log_2(n)$ . As an example, consider 3 frequency bands in figure 2 that are constructed from a set of 8 samples. 4 coefficients are created in the highest frequency band, 2 coefficients in the mid-band and a single coefficient in the lowest frequency band. A scale variable is always left out. This variable is needed for a possible reconstruction of the signal.

This transformation could also be viewed as passing the signal through a high pass and a low pass filter where the high pass filter retains the coefficients and the output from the low pass filter is further iterated. The frequency bands are calculated by using Nyquist's sampling theorem (Nyquist 1928), later proven in (Shannon 1949). The highest frequency band will have the interval of  $\frac{f_s}{2} - \frac{f_s}{4}$ , where  $f_s$  is the sampling frequency. The upper bound is due to Nyquist as seen in equation 1, and the lower bound due to the high pass filter (only the upper half of the frequencies can be detected). The range  $\frac{f_s}{4} - 0$  are a part of the scale variables that passed the low pass filter. These variables will be the samples to the next iteration, where they will be divided into  $\frac{f_s}{4} - \frac{f_s}{8}$  for the high pass filter, and  $\frac{f_s}{8} - 0$  for the low pass filter. By using a sample frequency of 4 Hz as an example in figure 2, the high frequency band interval would be  $1 - 2Hz$ , the mid  $0.5 - 1Hz$  and the low  $0 - 0.5Hz$ .

$$Nyquist : f_{max} = \frac{f_s}{2} \quad (1)$$

Nyquist Sampling Theorem, where  $f_s$  is the sampling frequency and  $f_{max}$  is the maximum detectable frequency within the signal.

The complexities of the investigated DWTs in this paper are in the order of  $O(\log_2(n))$ , hence within the boundaries of  $O(n)$ . Chan and Fu has already established this for the Haar DWT in (Chan & Fu 1999), however it that also applies to the D4 since it has the same complexity, compared to the  $O(n \log n)$  complexity of the FFTs (Sterns 2003). We investigate two DWT functions in this paper, the Haar and the Daubechies D4. The Haar was chosen because it has already been investigated in retrieval, and it is the simplest DWT. The D4 was chosen because it overcomes some limitations of the Haar.

### Haar

The Haar (Haar 1910) DWT is quite straight forward. It computes on pairs of samples. The function uses a pair of samples as the input and result in a scale variable and a coefficient. The scale is the averaged value of the two samples used in the pair, given by

$$scale(i) = \frac{sample(2i) + sample(2i + 1)}{2} \quad (2)$$

and the coefficient is the average change between the two samples given by

$$coef(i) = \frac{sample(2i) - sample(2i + 1)}{2} \quad (3)$$

This requires the Haar to have an input of only two samples. A matrix illustrating the Haar DWT is shown in equation 4.

$$\begin{pmatrix} s' \\ c' \end{pmatrix} = \begin{pmatrix} 0.5 & 0.5 \\ 0.5 & -0.5 \end{pmatrix} \times \begin{pmatrix} s_0 \\ s_1 \end{pmatrix} \quad (4)$$

$s_0$  and  $s_1$  are the input samples, or previously calculated scale variables.  $s'$  is the new scale variable calculated by the DWT.  $c'$  is the frequency coefficient, i.e., the difference between the two input samples. The Haar DWT has, as mentioned earlier, already been investigated in (Chan & Fu 1999). They argue that the Euclidian distance is not preserved unless the function is normalised by  $\frac{1}{\sqrt{2}}$  instead of  $\frac{1}{2}$ . A limitation with the Haar is that it mean-averages only two samples. As an example, consider a signal with the following sample sequence (5,5,25,25,5,5,25,25), where samples oscillate between two states which is not uncommon in bio-medical non-stationary signals. The effect of a Haar DWT would be to miss out on the high frequency oscillations. This is illustrated in figure 3.

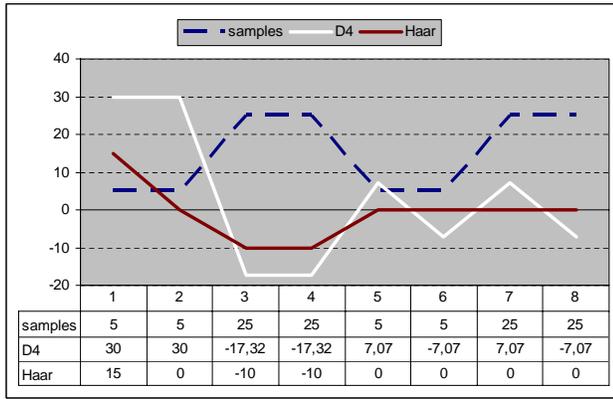


Figure 3: An example where Haar has problems in detecting oscillations in a signal.

The Haar DWT has sometimes problems with detecting changes in the input samples. As we can see, the Haar reports zero frequency changes in the entire high band (the rightmost 4 samples). The D4 wavelet detects the changes better than the Haar.

### Daubechies

The D4 DWT (Daubechies 1990) was introduced by Daubechies in the late 80's. The D4 overcomes the Haar's limitations by using 4 samples to calculate the scale and the coefficient. The D4 DWT matrix is

$$\begin{pmatrix} s' \\ c' \end{pmatrix} = \begin{pmatrix} h_0 & h_1 & h_2 & h_3 \\ h_3 & -h_2 & h_1 & -h_0 \end{pmatrix} \times \begin{pmatrix} s_0 \\ s_1 \\ s_2 \\ s_3 \end{pmatrix} \quad (5)$$

where

$$\begin{aligned} h_0 &= \frac{1+\sqrt{3}}{4\sqrt{2}} ; & h_1 &= \frac{3+\sqrt{3}}{4\sqrt{2}} \\ h_2 &= \frac{3-\sqrt{3}}{4\sqrt{2}} ; & h_3 &= \frac{1-\sqrt{3}}{4\sqrt{2}} \end{aligned}$$

and  $s_0 - s_3$  are the input samples.  $s'$  is the new scale; and  $c'$  is the frequency coefficients.

The D4 steps two samples per calculation in the sequence of samples even though it uses four samples as an input. This causes a small problem when there are only two samples left in the sequence. This problem is often addressed by making the sequence longer. This can be done either by making the sequence cyclic (start over with the first two samples) or by mirroring the sequence.

### Overcome the $2^n$ boundary

A drawback of DWTs is that they are only able to process sequences with sample lengths of  $2^n$ . The solution is to pad, i.e., add something to the sequence without altering the information the signal carries.

We solve this by mean value the signal, i.e., lower (or raise) the signal to oscillate around zero, as seen in equation 6. Once the sequence is mean valued, the sequence may be padded with zeroes up to the nearest  $2^n$  boundary, without introducing any artifacts.

$$sample(i)_{i=1}^n = sample(i) - \frac{\sum_{j=1}^n sample(j)}{n} \quad (6)$$

### Weights

We adopted, and adapted, the weighting scheme from the D-HS<sup>T</sup> for our method. The sequence is normalised in the range of -1 to 1 by the same means as in equation 6. The normalised frequency coefficients are placed in slots

$$slot(c) = \begin{cases} \Theta - 1 & \text{iff } c = 1 \\ \lfloor \Theta c \rfloor & \text{iff } c < 1 \end{cases} \quad (7)$$

where  $\Theta$  is the number of slots that the normalised interval is supposed to be divided into, and  $slot$  is the specific slot a frequency coefficient  $c$  is associated with.

Each frequency band is assigned a weight as described by

$$weight(\vartheta) = \begin{cases} 1.0 & \text{iff } \vartheta = 0 \\ \frac{1}{2^\vartheta} & \text{iff } \vartheta > 0 \end{cases} \quad (8)$$

$\vartheta$  is the sub band, i.e., frequency band, and  $weight(\vartheta)$  is the weight for the coefficients in the sub band. Thus, the highest frequency band, as in the example in figure 2, has the weight 1.0, the mid-band has weight 0.5 and the third and lowest band has 0.25 as its weight.

## Retrieval

We apply the weights to the entire set of coefficients in the sequence. We use two different similarity methods depending on whether the sequences have the same length or not. If two sequences has the same length they can be compared with a straight forward method of comparing coefficients. If two coefficients are in the same slot, they are said to be similar, as seen in equation 9 and 10. Similarity for two coefficients is calculated by

$$match(a, b) = \begin{cases} weight & iff \quad slot(a) = slot(b) \\ 0 & iff \quad slot(a) \neq slot(b) \end{cases} \quad (9)$$

The function  $match(a, b)$  returns the *weight* if sample  $a$  and  $b$  are similar. *weight* is the local weight of the sub band.  $slot()$  is given by equation 7. Similarity for the entire sequence is calculated by

$$S = \sum_{i=1}^n eq(C_a(i), C_b(i)) \quad (10)$$

where  $S$  is the total similarity, i.e., similarity for all coefficients in the sequence.  $C_a(i)$  is coefficient  $i$  in sequence  $a$ .  $C_b$  is sequence  $b$ , and  $eq()$  is equation 9. An example is illustrated in figure 4. Here are two sequences, each carrying 7 coefficients. We can see that  $C'_0$  and  $C''_0$  matches the same area in both time and frequency band. If the coefficients  $C'_0$  and  $C''_0$  are in the same slot (equation 7), they are deemed equal, and the weight (in this case 1.0) is added to the similarity.

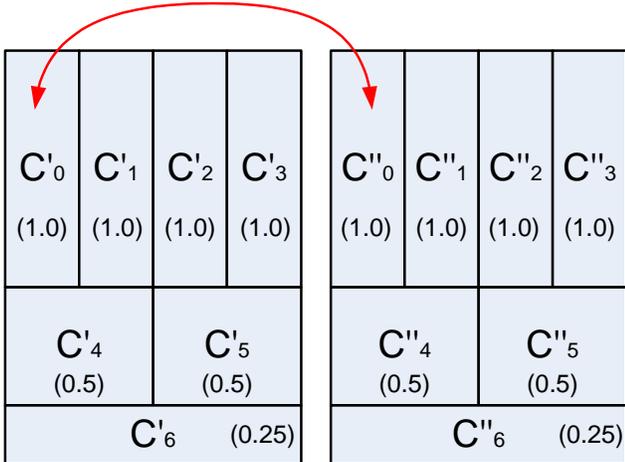


Figure 4: Frequency coefficients on sequences with the same length are matched.

There are three schemes of comparing sequences of different lengths. We call these schemes best fit, worst fit and averaged fit. In the case of best fit, we over fit the sequences

to have a less or equal distance between the sequences. The other two, worst and averaged fit is the "what if" approaches, where we under fit the similarities and see what happens.

**Best fit** The best fit uses the best match between coefficients representing the same sub band and relative time span. Figure 5 illustrates the comparison. Here  $C'_0$  has to be matched against both  $C''_0$  and  $C''_1$  as they cover the same area of scale and relative time. The best fit returns a match if either  $C''_0$  or  $C''_1$  are in the same slot as  $C'_0$ . There is an over fit if only one of  $C''_0$  or  $C''_1$  are in the same slot as  $C'_0$ . The weight is returned as a measurement of similarity, in this case 1.0.

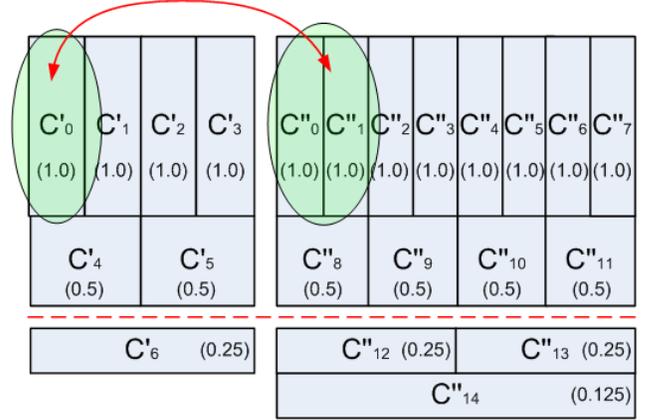


Figure 5: The best fit scheme where sequences with dissimilar lengths are compared.

An additional issue is the handling of sub bands that span different frequencies. In figure 5,  $C'_6$  is actually spanning the same frequency band as  $C''_{12}$  and  $C''_{14}$  combined. There are several ways of dealing with this issue. The easiest is to ignore everything below the red line in the figure, i.e., coefficients  $C'_6$ ,  $C''_{12}$ ,  $C''_{13}$  and  $C''_{14}$ . The second one is to compare  $C'_6$  with  $C''_{12}$ ,  $C''_{13}$  and ignore the fact that they do not belong to the same sub band. The third and final is to add the coefficient in the lowest sub band to its upper neighbouring sub band, i.e., we add the value of coefficient  $C''_{14}$  to coefficient  $C''_{12}$  and  $C''_{13}$ .  $C'_6$  is now comparable with  $C''_{12}$  and  $C''_{13}$  as they are all representing the same sub band. We opted for the last solution as we wanted to compare all frequency bands.

**Worst fit** Worst fit matches coefficients in the same manner as the best fit. The difference is that the worst fit scheme returns the worst result. If we use the example in figure 5, the worst fit will only return a match if both  $C'_0$  and  $C''_1$  are in the same slot as  $C'_0$ . Otherwise it is considered a miss and the similarity value 0 is returned.

**Average fit** The average fit calculates the average match of all coefficients in the area. In the case of figure 5

$$average = \frac{match(C'_0, C''_0) + match(C'_0, C''_1)}{2} \quad (11)$$

where  $slot()$  is given by equation 7.

## Test and Results

The test was conducted by using a set of cases. A case contains a class identifier and a set of heart rate (HR) samples. The cases are classified as belonging to one of 11 classes, the classes can be viewed in (von Schéele 1999). The time series in the cases, the sequences of HR, are measurements from clinical work, thus some are from normal healthy patients, and some are from patients suffering from different dysfunctions. These HR time series vary in length because they measure the heart activity during a breath, and a breath is approximately 3 – 15 seconds. The sampling frequency of the HR samples is 2 Hz, which leads us to expect sample lengths up to 30 – 40 samples. The number of cases in the test set is 642. Cases are divided in to two groups, cases for the case library and cases used in retrieval test based on the case library. Cases are semi-randomly fetched from the set of cases to the two groups. The semi-random function ensures that there is a good spread of cases from different patients, sessions and classes.

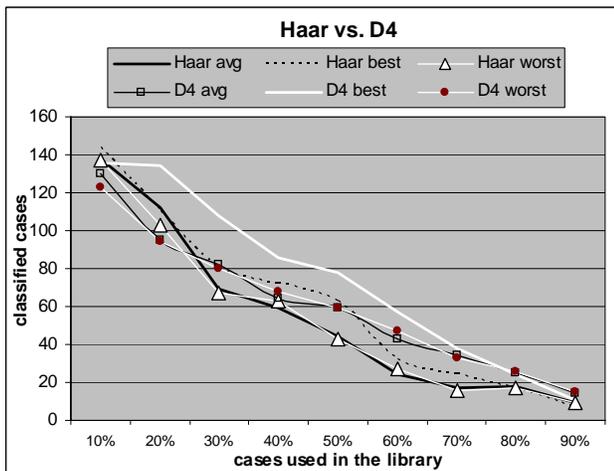


Figure 6: A comparison of Haar and D4 DWTs with different similarity schemes.

We tested the three similarity fitness schemes on both the Haar and the D4 DWT. In figure 6 a comparison of the retrieval performance between the Haar and the D4 DWT with different similarity schemes is shown. The horizontal axis denotes how many percent of the total amount of available cases that is used in the case library. The remaining cases are used in the retrieval test. 10% means that approximately 10% of the set of 642 cases is used in the case library, and that the remaining 90% is used in the retrieval test. Higher marks on the vertical axis equals more cases retrieved. As we can see, the D4 outperformed the Haar when compared with the same similarity schemes and the best fit similarity performed the best on both Haar and D4. The Haar loses against the D4 because its inability to pick up rapid changes within the signal, as described in

the wavelet section.

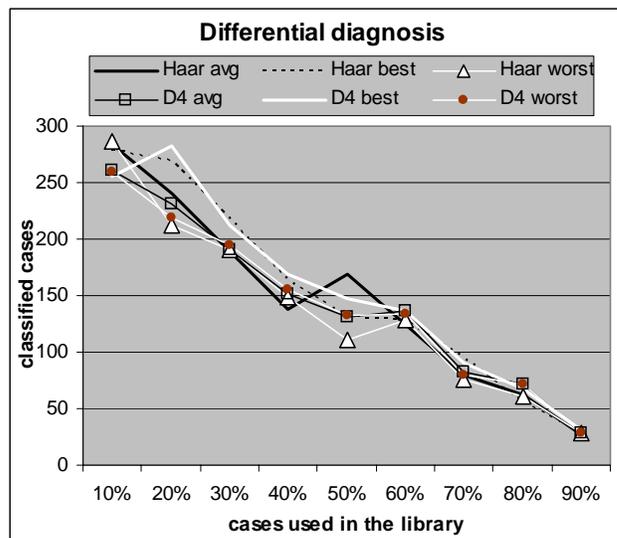


Figure 7: Differential diagnostics test where the first three suggested classes in a match counts as being similar.

In the second test we tested differential diagnostics performance. Differential diagnostics is used to give clinicians a broader range of support by suggesting more than one case to compare to, as there might be a combination of factors in a final diagnosis. In this case, suggesting several classes. We suggest the 3 most similar classes in this test. If a case is ranked within the top 3 of its actual class, it is considered correctly classified. The results from the differential diagnostics test is shown in figure 7. The two DWTs with their different similarity schemes are a lot closer in performance in this test. The performance is basically the same for all, except that the D4 and the Haar with the best fit scheme is just ahead of the other schemes above 50%. The distance is closer because the probability of having a correct classification is increased by a factor of 3 for all DWT/scheme methods, which makes the weaker methods gain ground with several classification attempts as well as less cases to classify, as we can see at the end of the graph.

As a comparison, we tested the D- $HS^T$  and an ordinary DFT with 128 complex frequency coefficients against the DWTs with the best fit scheme. The DFT is implemented as an FFT. The similarity measure for the DFT is the Euclidian distance based on the energy of each frequency coefficient given by

$$dist = \sum_{i=1}^{128} |x_i| - |y_i| \quad (12)$$

where  $dist$  is the Euclidian distance between the vectors,  $x$  and  $y$ , who carries complex Fourier coefficients.

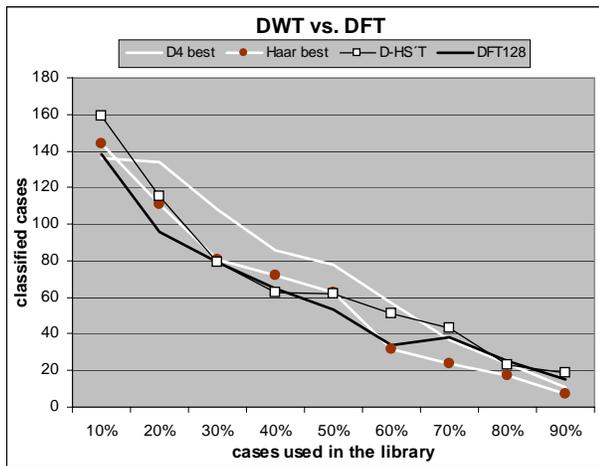


Figure 8: A comparison between time-frequency based transformation methods (DWTs) and pure frequency based methods (DFTs).

The D4 with the best fit scheme outperforms the DFTs, as seen in figure 8. Here the D4 with the best fit scheme shows improvement against the DFTs, but the Haar is slightly behind the D-HS<sup>T</sup>. However, the D-HS<sup>T</sup> is at par, or even outperforms our method in the beginning and in the end of the graph. This is probably due to the asymmetric distribution of cases in the test set. The cases are as earlier mentioned collected from measurement sessions from clinical every-day work. The more severe cases containing dysfunctions are more infrequent than common non-dysfunctional cases. The most common dysfunctions are often classifiable by using only the frequency spectrum. The equal performance between the DWTs and the DFTs could be that the equation for determining the number of cases from each class that is going to be in the case library is a ceiling function. That is, if there are 3 cases of a specific class in the set, and 70% of all cases are to be used in the case library, the function  $\lceil 0.7 * 3 \rceil$  returns 3. Hence, all 3 cases are used in the case library and no cases are left to be matched against.

We also tried a non weighting scheme due to the fact that small oscillations might be picked up by the higher frequency bands and thus would be impossible to tell which frequency bands or parts of them that are more important than others. This scheme of a static weight of 1 for all coefficients did perform poor in the tests. The D4 with the best fit scheme is implemented in a decision support system, HR3Modul, for Respiratory Sinus Arrhythmia (Nilsson, Funk, & Xiong 2005); and it have proven to be successful. Furthermore, it would be quite interesting to see the validity of this approach when applied to other domains and their data sets.

## Conclusions

The DWT has a couple of advantages compared to the DFT. It is shown that a DWT is faster to compute than a DFT

as a D4 has the complexity in the order of  $O(n)$  whereas a FFT has  $O(n \log n)$ . The DWT is also preferable when analysing temporal attributes in signals as the DWT is more accurate in both time and frequency compared to DFT based time-frequency methods.

These facts make DWTs candidates for retrieving relatively short biomedical time series. In tests, the DWT outperformed the DFT. The best retrieval combination was found to be the D4 with a best fit similarity scheme. The best fit scheme over fits the similarity between two sequences in the time-frequency domain. The D4 with the best fit clearly outperformed other combinations accept in the differential diagnostics test where there basically was a tie between all retrieval methods.

## References

- Bentley, P. M., and McDonnell, J. T. E. 1994. Wavelet transforms: an introduction. *Electronics & communication engineering journal* 6(4):175–186.
- Chan, K.-P., and Fu, A. W.-C. 1999. Efficient time series matching by wavelets. In *ICDE*, 126–133.
- Daubechies, I. 1990. The wavelet transform, time-frequency localization and signal analysis. *IEEE transactions on information theory* 36(5):961–1005.
- Haar, A. 1910. Zur theorie der orthogonalen funktionensysteme. *Mathematische Annalen* 69:331–371.
- Hippenstiel, R. D. 2002. *Detection Theory*. 2000 N.W. Corporate Blvd., Boca Raton, Florida 33431: CRC Press.
- Nilsson, M., and Funk, P. 2004. A case-based classification of respiratory sinus arrhythmia. In *Proceedings of the 7th European Conference on Case-Based Reasoning*, 673–685.
- Nilsson, M.; Funk, P.; and Xiong, N. 2005. Clinical decision support by time series classification using wavelets. In *7th International Conference on Enterprise Information Systems*. ICEIS'05.
- Nyquist, H. 1928. Certain topics in telegraph transmission theory. *Transaction on AIEE* 47:617–644.
- Oppenheimer, A., and Schafer, R. 1975. *Digital Signal Processing*. Englewood Cliffs, N.J.: Prentice-Hall.
- Patterson, D.; Galushka, M.; and Rooney, N. 2004. An effective indexing and retrieval approach for temporal cases. In *Proceedings of the 17th International FLAIRS Conference*, 190–195.
- Shannon, C. E. 1949. Communication in the presence of noise. *Proceedings of Institute of Radio Engineers* 37(1):10–21.
- Smith, S. W. 1999. *The Scientist and Engineer's Guide to Digital Signal Processing*. San Diego, California: California Technical Publishing.
- Sterns, S. A. 2003. *Digital Signal Processing with Examples in Matlab*. CRC Press.
- von Schéele, B. 1999. *Classification Systems for RSA, ETCO2 and other physiological parameters*. Heden 110, 821 31 Bollnäs, Sweden: PBM Stressmedicine.