International Journal of Software Engineering and Knowledge Engineering
© World Scientific Publishing Company

TIME SERIES CLASSIFICATION WITH DISCRETE WAVELET TRANSFORMED DATA

DAOYUAN LI, TEGAWENDE F. BISSYANDE, JACQUES KLEIN, and YVES LE TRAON
Interdisciplinary Centre for Security, Reliability and Trust
University of Luxembourg
4, rue Alphonse Weicker, L-2721 Luxembourg, Luxembourg
{*daoyuan.li, tegawende.bissyande, jacques.klein, yves.letraon*}*@uni.lu*

Received (Day Month Year)
Revised (Day Month Year)
Accepted (Day Month Year)

Time series mining has become essential for extracting knowledge from the abundant data that flows out from many application domains. To overcome storage and processing challenges in time series mining, compression techniques are being used. In this paper, we investigate the loss/gain of performance of time series classification approaches when fed with lossy-compressed data. This extended empirical study is essential for reassuring practitioners, but also for providing more insights on how compression techniques can even be effective in smoothing and reducing noise in time series data. From a knowledge engineering perspective, we show that time series may be compressed by 90% using discrete wavelet transforms and still achieve remarkable classification accuracy, and that residual details left by popular wavelet compression techniques can sometimes even help achieve higher classification accuracy than the raw time series data, as they better capture essential local features.

Keywords: Time Series Classification; Dimensionality Reduction; Discrete Wavelet Transform.

1. Introduction

Time series data exists in numerous application domains within our daily life. Especially, as the concept of pervasive computing and Internet of Things (IoT) slowly becomes reality, time series data are generated at industry scale. For instance, the BLUED non-intrusive load monitoring dataset [1] has been collected from a single household for one week, recording voltage and current measurements with a sampling rate of 12 kHz, leading to a total of tens of billions of time series readings and making it difficult to learn meaningful patterns in real-time. Another power company in the US records a trillion data points every four months; and in astronomy satellites may collect one trillion data points of starlight curves every single day [2]. Other domains where time series are prevalent include financial applications (e.g. currency exchanges and stock prices), environment monitoring (e.g. weather forecast and disaster monitoring), medical and health care (e.g. electrocardiograms a.k.a. ECGs). Besides its abundance characteristics, time series data are known for its extremely high dimensionality, often making general-purpose machine learning algorithms fail or underperform. As a result, in order to learn meaningful knowledge from time series data, efficient machine learning algorithms for time series are desired.

Extracting useful knowledge from time series data – a.k.a. time series mining [3] – is now popular but remains a challenging research topic. In particular, time series classification (TSC) attracts significant interest among the researchers and industry practitioners. TSC approaches often implement supervised learning techniques to classify unknown time series instances based on knowledge gained from existing labeled ones. Due to the abundance and intrinsic high dimensionality of time series data, computing resources such as storage, CPU and memory have become critical bottlenecks in the exploitation of time series. To address these challenges, the research community has proposed efficient dimensionality reduction [4] and representation mechanisms such as SAX [5] and Discrete Wavelet Transforms (DWT). DWT is very popular among both researchers and industry practitioners. In general, wavelets are mathematical functions that process raw data to only keep meaningful oscillations in data values. The earliest wavelet was brought up by Haar in 1909, but has undergone great development especially since Ingrid Daubechies [6] proved the existence of wavelet families with compact support over an interval [7] and orthogonal translates [8]. Wavelets compression techniques have since then been extensively used in image compression and are part of the JPEG 2000 standard [9]. Wavelet transforms have also been extensively used in medical data analysis including ECG diagnosis [10] and ultrasound image processing [11]. More recently, researchers have applied wavelet transforms in the field of Non-Intrusive Load Monitoring (NILM) [12, 13].

Although wavelets are popular in the research community, the literature lacks a large scale empirical study on the impact of wavelet transformation on the performance of time series mining approaches. In this paper, which extends our previous work [14], we seek to investigate this impact on a baseline state-of-the-art time series classification approach. To that end, we compare the classification accuracy of raw uncompressed time series data against transformed data using DWT, including both wavelet approximations and the details, i.e., the residuals or *noises* that are often thrown away during the lossy compression processes. In this way, we are able to separate time series' global features from local defining subsequences. Furthermore, through extensive significance tests, we prove that wavelets can perform better than explicit smoothing techniques in TSC tasks.

In this study, we extensively test how discrete wavelet transforms impact TSC accuracy and computational efficiency using 39 openly accessible datasets. Our study suggests that DWT can indeed be useful in time series classification tasks:

- Wavelet transforms can be used to reduce dimensionality of time series data, while at the same time achieving similar classification accuracy compared to using the original uncompressed data. In fact, we demonstrate that time series dimensionality may be reduced by around 90% while still achieving good classification accuracy.

- Wavelets can be used to reduce noises in time series data, so that better classification performance can be achieved after conducting wavelet transform on the original uncompressed data.
- Wavelets implicitly smoothens time series data, and they can be slightly more effective for time series classification compared to explicit smoothing techniques.
- We have further found that, surprisingly, for a few datasets, classification using the compression residual details can be even more effective than using either the original data or the wavelet approximation. This finding suggests that there are specific datasets which are more distinguishable using local features instead of global ones.

The remainder of this paper is organized as follows. We introduce the necessary background information in Section 2 and related work in Section 3. We present experimental details and results in Section 4 and study the implicit smoothing effect of wavelets in Section 5, before concluding the paper and outlining future work in Section 6.

2. Background

In this section, we present the necessary background to facilitate understanding of this paper. Specifically, we firstly introduce time series and the baseline classification approach used in this paper. Then we show how DWT works and especially how DWT can be applied to time series dimensionality reduction.

2.1. *Time Series and TSC*

Generally, time series is one important type of temporal data. In the data mining community, time series data are often referred to as ordered lists of numeric values [15]. In this paper, a time series $T = t_0, t_1, ..., t_{n-1}$, where t_i $(0 \leq i \leq n-1)$ is a finite number and T has a length of n, i.e., $|T| = n$.

Time series classification (TSC) is a common category of tasks that involves learning from existing time series instances (training set) and applying the learned knowledge to assign labels to instances from a testing dataset, where instance classes or labels are often unknown (either this information does not exist or has been intentionally hidden from the classification process). TSC tasks are especially common in application domains such as image and speech recognition (e.g., for recognizing spoken words), medical diagnosis (e.g., for detecting the type of a heart disease in an ECG signal), gesture detection, and so on. Due to its numerous application scenarios, many techniques have been proposed for TSC, including k-Nearest Neighbors, shapelets [16], and bag-of-features [17]. Among them, the Nearest Neighbor

(1NN) approach has been proven to work exceptionally well, especially when using Dynamic Time Warping (DTW) [18] for computing the distance metric between a pair of time series samples. Fig. 1 shows an example of how DTW works. Unlike Euclidean distance where two time series are aligned point by point, i.e., the i^{th} point in series X is compared against the i^{th} point in another series Y, DTW tries to find the best way to *warp* the time axis and as a result aligns X and Y differently. As shown by the gray dotted lines in Fig. 1, an i^{th} point in X can be mapped to a j^{th} point (it is possible that $i \neq j$), or one point in X may even be mapped to multiple points in Y. For Euclidean distance, the gray dotted lines would all be vertical. Thanks to DTW's capability to describe similarities, we consider DTW-based 1NN classification as a reference TSC approach for investigating wavelet transformed time series data.

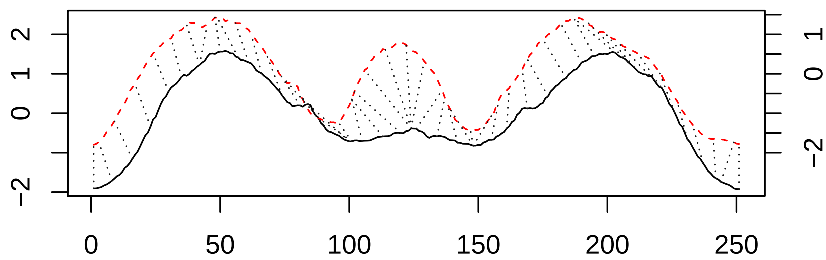

Fig. 1. Illustration of how DTW aligns two time series and calculates their distance.

2.2. *Discrete Wavelet Transforms*

Wavelets are mathematical functions that resemble the shape of wave oscillations, with the constraint for waves to start at 0 and then oscillate to 0 in the end. Mathematically, wavelets are described using two types of functions: the wavelet function (the mother function denoted with $\psi(t)$) and the scaling function (the father function denoted with $\phi(t)$) [19]. These functions have to satisfy the following conditions:

$$||\psi(t)||^2 = \int |\psi(t)|^2 dt < \infty \tag{1}$$

$$\int |\psi(t)| dt < \infty \tag{2}$$

$$\int \psi(t) dt = 0 \tag{3}$$

$$\int \phi(t) dt = 1 \tag{4}$$

The earliest and simplest wavelet function is Haar, whose wavelet function and scaling function are defined as follows:

$$\psi_{Haar}(t) = \begin{cases} -1 & 1/2 \leq t < 1, \\ 1 & 0 \leq t < 1/2, \\ 0 & \text{otherwise.} \end{cases}$$

$$\phi_{Haar}(t) = \begin{cases} 1 & 0 < t < 1, \\ 0 & \text{otherwise.} \end{cases}$$

As illustrated in Fig. 2, the Haar wavelet is not a very smooth one, that is, it has a low regularity. Other more sophisticated wavelets – for instance, Daubechies 20 and Symlets 20 – have a higher regularity and thus are able to represent signals more accurately.

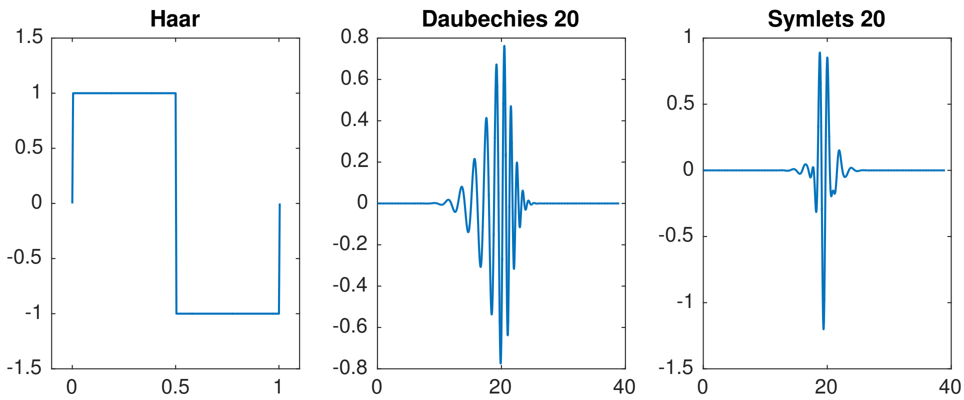

Fig. 2. Wavelet functions of Haar, Daubechies 20 and Symlets 20.

In practice, the Haar transform is performed as follows. Given a series $T = t_1, ..., t_n$, the Haar transform outputs two series: the approximation A and the details D, where for $1 \leq i \leq \frac{n}{2}$,

$$A_i = \frac{t_{2i-1} + t_{2i}}{\sqrt{2}} \tag{5}$$

$$D_i = \frac{t_{2i-1} - t_{2i}}{\sqrt{2}} \tag{6}$$

It is clear that the approximations capture the overall *shape* – the global features – of the original series, while the details are the variances – local features – in time series. Fig. 3 demonstrates an example of single level, one dimensional Haar transform. As shown, the original signal on the top has been compressed to half of the original size (figure in the middle, note the x-axis label) and the amplitude

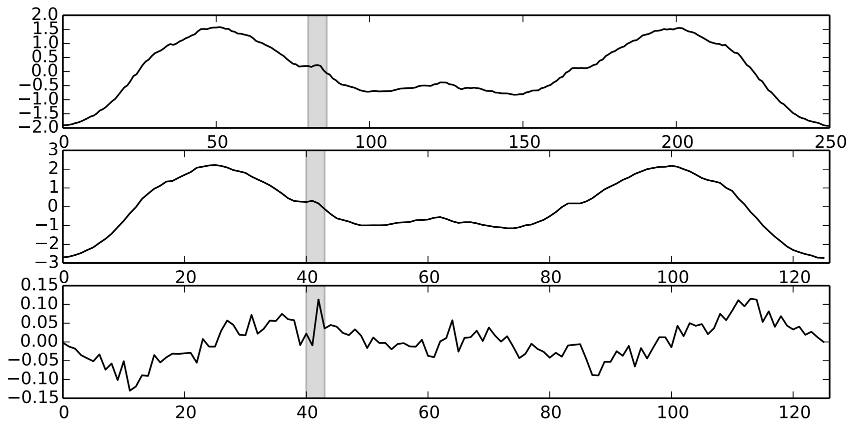

Fig. 3. Example of Haar transform: the original signal, the Haar approximation and the residual details.

has been scaled up despite that the overall signal shape is not changed. Note also the grayed area in the signal, it is clear that the approximation becomes smoother after transformation, and that the details shown in the bottom have been dropped. Note also that the approximation becomes smoother, i.e., after residual details are removed, the data points in the time series approximation do not tend to change drastically over time when comparing with the original time series. We will study the smoothing effect of Wavelets later in Section 5.

3. Related Work

Thanks to wavelets' wide application domains [20], there has been research investigating their performance in various domains. However, to the best of our knowledge, impact of different DWT techniques on TSC has not been done in a generic manner. As a result, in this section we enumerate relevant work to ours in the domains of medical applications, image compression, NILM, and so on.

Addison [10] has conducted a review of both continuous wavelet transform (CWT) and discrete wavelet transform (DWT) on ECG data, concluding that DWT is practically easy to use, while CWT – being more complex and difficult to tune parameters – is able to keep a high resolution in the time-frequency plane, which can result in more accurate identification of components. Another study in wavelet applications in the medical field was done by Pizurica et. al. [11], where the authors reviewed the performance of wavelet denoising specifically in MRI and brain imaging. Amolins et. al. [19] has compared wavelet transforms' performance in image fusion against standard fusion techniques including Intensity-Hue-Saturation (IHS) and principal component analysis (PCA), and found out that even the simplest wavelet-based scheme outperforms IHS and PCA.

Zhu et. al. [21] claim that wavelets are superior to Fourier methods in the field of tool condition monitoring in terms of signal denoising and feature extraction. To disaggregate electric signals, Duarte et. al. [12] take advantage of CWT in order to extract features from voltage transients and use the extracted features for classification. Gray et. al. [13] use wavelet-based classification for NILM and claim that "symlets behave in an identical manner as their less symmetric Daubechies representation". And as our empirical study shall demonstrate in the next section, Symlets can actually perform significantly better than Daubechies.

Chan et. al. [22] have proposed using Haar for more efficient similarity search, but have not considered other wavelet families. Finally, this work was partly inspired by our previous work, where we have taken advantage of SAX to transform/compress time series data and then build per-class language models to profile household electric appliances [23] and conduct general-purpose time series classification [24, 25]. When classifying, we compare time series against models instead of known samples. Especially, we have used SAX to conduct dimensionality reduction before converting real-valued time series into strings of alphabets. And our empirical experiments have suggested that dimensionality reduction can be pushed more using DWT, thus we provide such results in this study.

4. Experimental Study

In this section, we present our experimental setup and the collected results. In order to facilitate reproducibility, we opt to experiment on publicly available datasets, and further open source our own implementation[a].

4.1. *Setup and Datasets*

The goal of our study is to investigate how wavelet transformed data may impact TSC performance. Therefore, we do not intend to compare the performance of different classifiers and similarity metrics (which have been empirically studied in [26]). We rely in our study on the most frequently used classification method: Nearest Neighbor Classification (1NN) with DTW distance. In this study we choose to calculate the classification performance first in Section 4.2, Section 4.3 and Section 4.4 using FastDTW [27], which is a DTW approximation with linear time complexity, and then extend the experiments using an exact DTW implementation named UCRSuite [2] in Section 4.5.

The datasets that we have experimented on are from the UCR Time Series Classification Archive [28], which contains datasets from various domains ranging from electricity readings and medical signals such as Electrocardiographs (ECGs) to image recognition data. We have specifically chosen to use the *Newly Added Datasets*, which contains 39 separate datasets from various domains, with all these

[a]https://github.com/serval-snt-uni-lu/wavelets-tsc

Table 1. Characteristics of datasets used in our study.

#	Dataset Name	#Classes	#Training	#Testing	Length
1	ArrowHead	3	36	175	251
2	BeetleFly	2	20	20	512
3	BirdChicken	2	20	20	512
4	Computers	2	250	250	720
5	DistalPhalanxOutlineAgeGroup	3	139	400	80
6	DistalPhalanxOutlineCorrect	2	276	600	80
7	DistalPhalanxTW	6	139	400	80
8	Earthquakes	2	139	322	512
9	ECG5000	5	500	4500	140
10	ElectricDevices	7	8926	7711	96
11	FordA	2	1320	3601	500
12	FordB	2	810	3636	500
13	Ham	2	109	105	431
14	HandOutlines	2	370	1000	2709
15	Herring	2	64	64	512
16	InsectWingbeatSound	11	220	1980	256
17	LargeKitchenAppliances	3	375	375	720
18	Meat	3	60	60	448
19	MiddlePhalanxOutlineAgeGroup	3	154	400	80
20	MiddlePhalanxOutlineCorrect	2	291	600	80
21	MiddlePhalanxTW	6	154	399	80
22	PhalangesOutlinesCorrect	2	1800	858	80
23	Phoneme(readme)	39	214	1896	1024
24	ProximalPhalanxOutlineAgeGroup	3	400	205	80
25	ProximalPhalanxOutlineCorrect	2	600	291	80
26	ProximalPhalanxTW	6	205	400	80
27	RefrigerationDevices	3	375	375	720
28	ScreenType	3	375	375	720
29	ShapeletSim	2	20	180	500
30	ShapesAll	60	600	600	512
31	SmallKitchenAppliances	3	375	375	720
32	Strawberry	2	370	613	235
33	ToeSegmentation1	2	40	228	277
34	ToeSegmentation2	2	36	130	343
35	UWaveGestureLibraryAll	8	896	3582	945
36	Wine	2	57	54	234
37	WordSynonyms	25	267	638	270
38	Worms	5	77	181	900
39	WormsTwoClass	2	77	181	900

datasets sharing a unified file format and internal representation structure, which is convenient for batch processing. The UCR archive provides both the datasets and the ground-truth, i.e., the correct label of each testing instance.Table 1 summarizes the characteristics of these 39 datasets. These characteristics include the number of classes in the training and testing sets, how many instances are there in training and testing sets respectively, and finally the lengths (dimensionality) of time series samples. As shown, this archive comes with predefined training and testing sets, which makes it easier for researchers to compare results in an uniformed manner.

However, we are not aware of more detailed information about each dataset such as the sampling frequency and original amplitude, making it difficult for us to draw conclusions about domain-specific tasks.

Besides, it is obvious that several datasets are large in size, for instance, to classify all $7,711$ testing instances using 1NN in *ElectricDevices* (#10), there will be $7,711 * 8,926 = 68,828,386$ pairwise comparisons, making the classification process extremely time consuming. Due to the large amount of computation tasks, we have split the test datasets in order to parallelize computation. All the classification tasks are conducted on an HPC platform [29].

4.2. *TSC with Wavelet Transformed Data*

As a first step, we seek to investigate how DWT compressed data will impact classification accuracy compared with using raw uncompressed data. Here we transform all the 39 datasets using wavelets from seven well-known families, choosing the single wavelet with the highest regularity from each family. Concretely, the wavelets are: Haar, Daubechies 20, Symlets 20, Coiflets 5, Biorthogonal 6.8, Reverse biorthogonal 6.8 and Discrete Meyer with finite impulse response (FIR) approximation. In this step, all time series – both training instances and testing instances – from each dataset are processed using single level, one dimensional DWT. After transformation, the size of *compressed* data is reduced by half from the original series. Then, we use FastDTW-based 1NN to classify all the testing instances in each dataset. Thanks to the $O(n)$ time complexity of FastDTW, reducing time series sizes by half means reducing classification time by half. And for DTW with $O(n^2)$ complexity, classification time can be reduced by 75%.

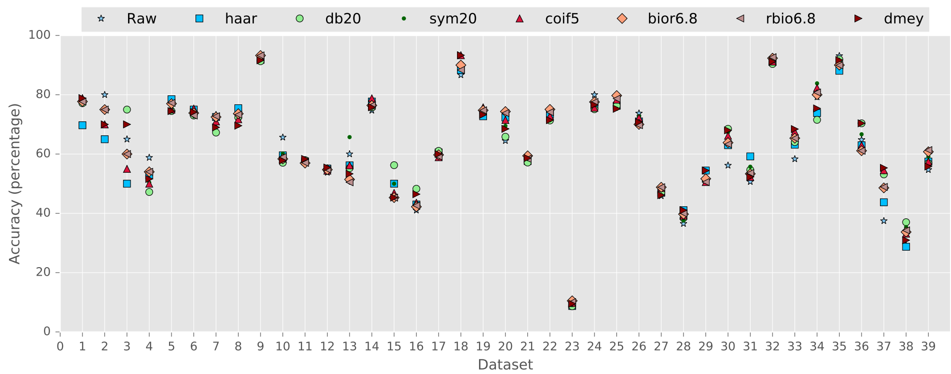

Fig. 4. Classification accuracy with FastDTW based 1NN, using original and DWT transformed/compressed data.

Fig. 4 presents the classification accuracy of each wavelet transformed dataset together with that of the original data. We note that, while the compression yields smaller datasets and leads to faster classification, it does not impact the classification accuracy in most cases. Furthermore, in the case of several datasets, the classification accuracy has actually been improved on compressed data. These observations suggest that wavelet transformations are indeed relevant means for noise reduction in TSC tasks.

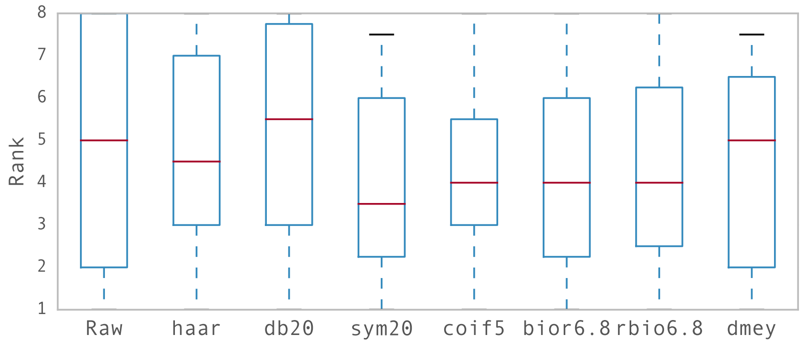

Fig. 5. Rank of classification accuracy by approximation of different wavelet transformation.

To further investigate which wavelet family performs better globally, we rank each wavelet family's classification performance per dataset and draw a boxplot chart of these rankings. As shown in Fig. 5, in general wavelet transform data performs better compared with the original data, thanks to wavelets' noise reduction functionality. Regarding the performance of individual wavelets, **Symlets 20** generally outperforms the rest, including classification using original data. And much to our surprise, *Daubechies 20* in general performs the worst among all tested wavelets, indicating that the most smooth wavelet may not be the most suitable wavelet for TSC.

4.3. *TSC with Residual Details*

It is intuitive that when using wavelet transformation, the approximation of original data keeps more relevant information than the residual details. However, in the next experiments, we demonstrate that the residual details may be also useful for TSC and that in some scenarios the *noises* are more discriminative features than the approximations. To prove this seemingly counter-intuitive point, we follow the same procedures as Section 4.2 to compress all datasets, while instead of keeping the approximations, here we drop all of them and consider the residual details, i.e., the noises. Again, to be fair, we compare the classification accuracies using the noises against that using the original raw data.

Fig. 6 presents the classification accuracies using only the residual details. As shown, although the classification accuracies are generally lower when classifying using only the residuals, they are still surprisingly high in many cases. Especially, for several datasets, classification accuracy using only residuals are similar or higher than using the raw data. Next, we try to rank how discriminative the residuals

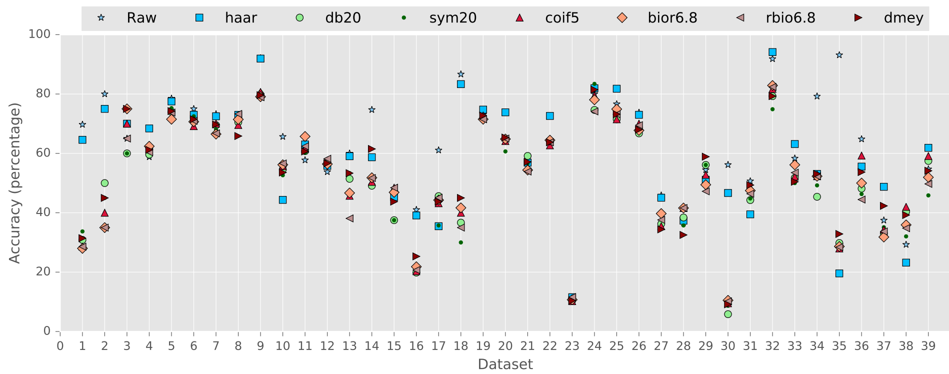

Fig. 6. Classification accuracy with FastDTW based 1NN, using original data and residual details from DWT transform.

from each wavelet family are. The boxplot shown in Fig. 7 suggests that residuals produced by *Haar* are most discriminative, being almost as good as the original data. *Symlets 20*, on the other hand, falls on the other extreme. These two observations are in accordance with the finding in Section 4.2, suggesting that *Symlets 20* is good at keeping globally relevant information during transformation. This observation suggests that residuals from DWT compression can also be useful, since these details contain local features that are potentially discriminative.

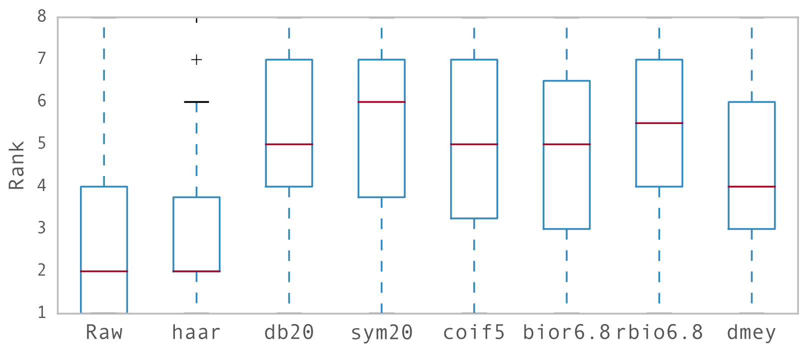

Fig. 7. Rank of classification accuracy by residual details from different wavelet transformation.

As a result, we believe that both the approximation and details transformed from the original data are important, due to the fact that the transforms extract two independent discriminative features. While some datasets are more distinguishable using global features, others are more so using local defining features.

4.4. *Multi-Level Wavelet Transformation*

So far we have only tested the performance of single level wavelet transformation. Since it is possible to conduct wavelet transformation in multiple levels, we seek to investigate how TSC performance can be affected when transforming data through multiple levels. To be specific, based on the good results in previous sections, we transform all datasets with *Symlets 20* from level 2 till the maximum level as permitted by the wavelet transform library[b]. That is, due to different time series lengths, even level 2 transformation is not possible for short time series, while longer time series may be processed using level 6 Symlets 20 decomposition. In Table 2 we present both the percentage of dimension/size reduction (**R**) and the corresponding classification accuracy (**A**) in each level. Note that not all datasets support at least level 2 decomposition due to short lengths, as a result, these datasets are omitted in this table.

Table 2. Classification accuracy with FastDTW based 1NN, using Symlets 20 multi-level decomposition.

	Raw	**Level 2**		**Level 3**		**Level 4**		**Level 5**		**Level 6**	
#		**R**	**A**	**R**	**A**	**R**	**A**	**R**	**A**	**R**	**A**
1	69.7	63.3	**78.3**								
2	80.0	69.3	65.0	80.9	65.0						
3	65.0	69.3	**70.0**	80.9	**80.0**						
4	58.8	71.0	58.4	82.8	55.2	88.8	50.0				
8	71.7	69.3	**78.6**	80.9	69.9						
11	57.7	69.2	56.5	80.8	**59.7**						
12	53.8	69.2	**55.9**	80.8	**62.0**						
13	60.0	68.2	**65.7**	79.6	53.3						
14	74.7	73.9	**80.0**	86.3	**79.9**	92.4	**79.0**	95.5	71.1	97.0	70.7
15	45.3	69.3	**53.1**	80.9	**50.0**						
16	41.0	63.7	**44.1**								
17	61.1	71.0	**64.0**	82.8	**63.2**	88.8	**64.3**				
18	86.7	68.5	**93.3**	79.9	**93.3**						
23	9.9	72.2	**12.3**	84.2	**12.4**	90.2	**11.0**				
27	45.9	71.0	45.3	82.8	41.9	88.8	39.2				
28	36.5	71.0	**36.8**	82.8	**38.4**	88.8	**40.0**				
29	54.4	69.2	**55.0**	80.8	51.7						
30	56.2	69.3	**72.7**	80.9	**72.2**						
31	50.7	71.0	**61.9**	82.8	**65.9**	88.8	**67.5**				
32	91.8	62.6	91.7								
33	58.3	64.6	**68.0**								
34	79.2	66.5	**81.5**	77.6	77.7						
35	93.1	72.0	92.4	83.9	**93.2**	89.9	91.5				
36	64.8	62.8	55.6								
37	37.5	64.4	**61.3**								
38	29.3	71.8	**35.9**	83.8	**45.3**	89.8	**47.0**				
39	54.7	71.8	**60.2**	83.8	**64.6**	89.8	**65.7**				

[b]PyWavelets – Discrete Wavelet Transform in Python (`http://www.pybytes.com/pywavelets/`)

As shown in Table 2, although many datasets are compressed by 80% to 90% in size, the classification accuracy using these reduced data can still outperform those using original compressed data. As a result, we think it is safe to claim that multi level wavelet transformation are indeed helpful for TSC tasks when it comes to classifying long time series. Note especially the *HandOutlines* (#14) dataset, due to its extremely high dimensionality, we are able to compress them by up to 97% of the original size, while still obtaining remarkably high classification accuracy. Since FastDWT normally has a time complexity of $O(n)$, this indicates huge time savings in the classification process.

4.5. *Using the UCR suite for TSC and Significance Test*

Since the distance measure we have used – FastDTW – is approximate [14], we extend our evaluation using an exact distance measure named UCRSuite [2], in order to get a more accurate view of DWT's performance in terms of noise reduction when it comes to classification. Note that since UCRSuite requires a DTW warping window size to be configured when calculating time series distances, we have tried all possible warping window size specified as a percentage $(0\%, 1\%, 2\%, ..., 99\%)$ of lengths of corresponding time series samples and select the best classification accuracy.

Besides using a more accurate distance measure, we also seek to find out if one wavelet family indeed significantly outperforms another. To that end we conduct a Nemenyi test [30], which is a post-hoc test that takes pairwise tests of performance and decides if these groups are statitically similar. Fig. 8 shows the test result in the form of a critical difference diagram, where average ranks of all examined approaches are presented and bold lines (insignificance lines) indicate groups of approaches which are not significantly different. As shown, using a more accurate DTW distance implementation shows similar results, indeed indicating that wavelets can be very useful for dimensionality reduction in TSC tasks. Especially, we note that performances of *Haar*, *Reverse biorthogonal 6.8*, *Symlets 20*, *Biorthogonal 6.8* and *Coiflets 5* are not significantly different than using uncompressed raw data. Furthermore, it is consistant with our previous experiments that *Daubechies 20* and *Discrete Meyer with finite impulse response (FIR) approximation* do not perform as well as the other wavelets.

5. The Smoothing Effect of Wavelets

We could see clearly from Fig. 3 that the Haar Wavelet approximation compression smoothens the overall curve while removing the details. In this section, we try to understand if the good performances of DWT on time series data is due to the implicit smoothing effect. One of the simplest explicit smoothing techniques is probably simple moving average, which is calculated by computing the unweighted mean over a specific number of time periods. Similar to Haar as specified in Eq. 5,

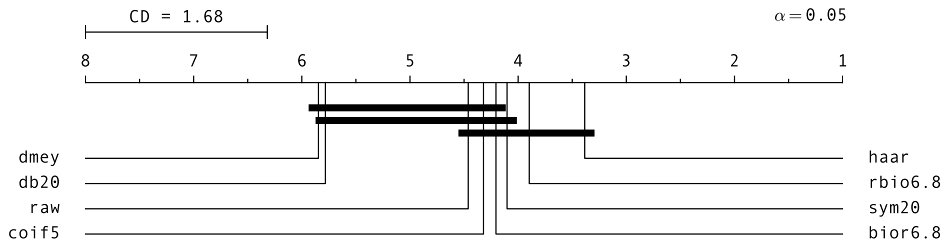

Fig. 8. Critical difference diagram for classification using raw and transformed data using different families of DWT.

simple moving average can be defined mathematically in Eq. 7:

$$SMA_i = \frac{\sum_{j=1}^{m} t_{m*i+j}}{m} \tag{7}$$

where $2 \leq m < \frac{n}{m}$ is the number of data points to average on.

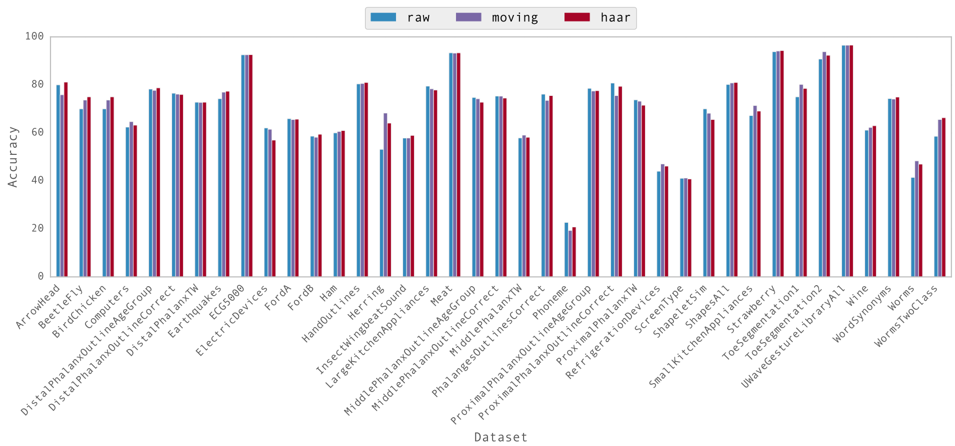

Fig. 9. Classification accuracy with UCRSuite DTW-based 1NN, using raw, Haar transformed and moving average smoothened data.

Due to their similarity, we first compare the performance of Haar against raw and moving average smoothened data. Fig. 9 illustrates the classification across 39 datasets used in the previous section. As shown, in some datasets Haar outperforms moving average (e.g., *ArrowHead* and *BeetleFly*), while in some others Haar underperforms (e.g., *ProximalPhalanxTW* and *ShapeletSim*). To answer the question whether wavelet transforms statistically outperform smoothing techniques, we need to run another significance test. To that end we compare DWT with explicit smoothing techniques, including moving average smoothing, local regression with

1st degree polynomial model (*lowess*) and second degree polynomial model (*loess*) as well as their robust versions (*rlowess* and *rloess* respectively), and finally the Savitzky-Golay filter (*sgolay*). These techniques are frequently used in real-world scenarios and are available from MATLAB®. Fig. 10 illustrates the significance test. We can see that overall wavelets perform better than explicit smoothing techniques. Especially, simple moving average smoothing helps but not in significant terms. Besides, *Haar*, *Reverse biorthogonal 6.8*, *Symlets 20*, *Biorthogonal 6.8* and *Coiflets 5* slightly outperforms moving average, indicating wavelets' superiority in general purpose time series data smoothing and noise reduction.

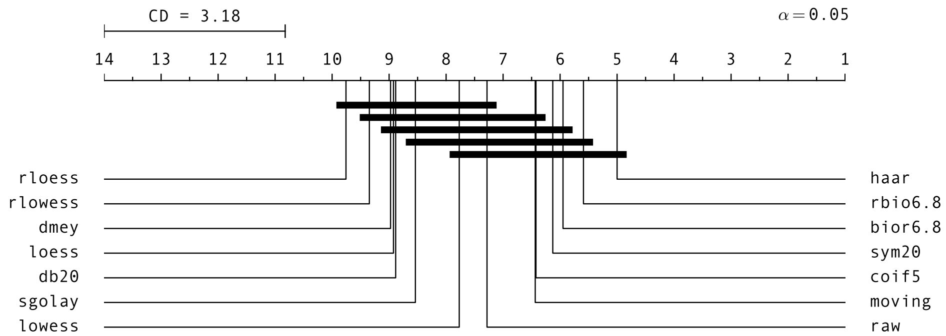

Fig. 10. Critical difference diagram for classification using raw and transformed data using different families of DWT and explicit smoothing techniques.

6. Conclusions and Future Work

Discrete Wavelet Transform techniques have matured in the past decades to deliver high data compression rates. Applied to time series data, existing DWT-based lossy compression approaches help to overcome the challenges of storage and computation time. In this paper, we provide assurances to practitioners by empirically showing with various datasets and with several DWT approaches that time series classification yields similar accuracy on both compressed (i.e., approximated) and raw time series data. We also show that, in some datasets, wavelets may actually help in reducing noisy variations which deteriorate the performance of mining tasks. In a few cases, we note that the residual details/*noises* from compression are more useful for recognizing data patterns.

In future work, we plan to extensively investigate the characteristics of time series datasets, in order to empirically correlate successful wavelets techniques per application domains. Dataset characterizations will also help identify which types of time series data can benefit from the use of residual details instead of the approximation data.

Acknowledgements

This work is supported by Paul Wurth S.A. and Luxembourg Ministry of Economy. The authors would also like to thank Prof. Eamonn Keogh for his concrete suggestions on an earlier version of this study.

References

[1] K. Anderson, A. Ocneanu, D. Benitez, D. Carlson, A. Rowe, and M. Berges, "Blued: A fully labeled public dataset for event-based non-intrusive load monitoring research," in *Proceedings of the 2nd KDD workshop on data mining applications in sustainability (SustKDD)*, 2012, pp. 1–5.

[2] T. Rakthanmanon, B. Campana, A. Mueen, G. Batista, B. Westover, Q. Zhu, J. Zakaria, and E. Keogh, "Searching and mining trillions of time series subsequences under dynamic time warping," in *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining.* ACM, 2012, pp. 262–270.

[3] T.-C. Fu, "A review on time series data mining," *Engineering Applications of Artificial Intelligence*, vol. 24, no. 1, pp. 164–181, 2011.

[4] Q. Wang and V. Megalooikonomou, "A dimensionality reduction technique for efficient time series similarity analysis," *Information systems*, vol. 33, no. 1, pp. 115–132, 2008.

[5] J. Lin, E. Keogh, L. Wei, and S. Lonardi, "Experiencing sax: a novel symbolic representation of time series," *Data Mining and knowledge discovery*, vol. 15, no. 2, pp. 107–144, 2007.

[6] I. Daubechies, "Orthonormal bases of compactly supported wavelets," *Communications on pure and applied mathematics*, vol. 41, no. 7, pp. 909–996, 1988.

[7] A. Cohen, I. Daubechies, and P. Vial, "Wavelets on the interval and fast wavelet transforms," *Applied and computational harmonic analysis*, vol. 1, no. 1, pp. 54–81, 1993.

[8] I. Daubechies, "Orthonormal bases of compactly supported wavelets ii. variations on a theme," *SIAM Journal on Mathematical Analysis*, vol. 24, no. 2, pp. 499–519, 1993.

[9] D. S. Taubman and M. W. Marcellin, "JPEG2000: Standard for interactive imaging," *Proceedings of the IEEE*, vol. 90, no. 8, pp. 1336–1357, 2002.

[10] P. S. Addison, "Wavelet transforms and the ECG: a review," *Physiological measurement*, vol. 26, no. 5, p. R155, 2005.

[11] A. Pizurica, A. M. Wink, E. Vansteenkiste, W. Philips, and B. J. Roerdink, "A review of wavelet denoising in MRI and ultrasound brain imaging," *Current medical imaging reviews*, vol. 2, no. 2, pp. 247–260, 2006.

[12] C. Duarte, P. Delmar, K. W. Goossen, K. Barner, and E. Gomez-Luna, "Non-intrusive load monitoring based on switching voltage transients and wavelet transforms," in *Future of Instrumentation International Workshop (FIIW), 2012*. IEEE, 2012, pp. 1–4.

[13] M. Gray and W. Morsi, "Application of wavelet-based classification in non-intrusive load monitoring," in *Electrical and Computer Engineering (CCECE), 2015 IEEE 28th Canadian Conference on.* IEEE, 2015, pp. 41–45.

[14] D. Li, T. F. Bissyandé, J. Klein, and Y. Le Traon, "Time Series Classification with Discrete Wavelet Transformed Data: Insights from an Empirical Study," in *The 28th International Conference on Software Engineering and Knowledge Engineering (SEKE 2016)*, July 2016, pp. 273–278.

[15] T. Rakthanmanon and E. Keogh, "Fast shapelets: A scalable algorithm for discover-

ing time series shapelets," in *Proceedings of the thirteenth SIAM conference on data mining*, 2013.

[16] L. Ye and E. Keogh, "Time series shapelets: a new primitive for data mining," in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2009, pp. 947–956.

[17] M. G. Baydogan, G. Runger, and E. Tuv, "A bag-of-features framework to classify time series," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2796–2802, 2013.

[18] G. E. Batista, X. Wang, and E. J. Keogh, "A complexity-invariant distance measure for time series." in *SDM*, vol. 11, 2011, pp. 699–710.

[19] K. Amolins, Y. Zhang, and P. Dare, "Wavelet based image fusion techniques – An introduction, review and comparison," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 62, no. 4, pp. 249–263, 2007.

[20] A. K.-m. Leung, F.-t. Chau, and J.-b. Gao, "A review on applications of wavelet transform techniques in chemical analysis: 1989–1997," *Chemometrics and Intelligent Laboratory Systems*, vol. 43, no. 1, pp. 165–184, 1998.

[21] K. Zhu, Y. San Wong, and G. S. Hong, "Wavelet analysis of sensor signals for tool condition monitoring: a review and some new results," *International Journal of Machine Tools and Manufacture*, vol. 49, no. 7, pp. 537–553, 2009.

[22] F. K.-P. Chan, A. W.-c. Fu, and C. Yu, "Haar wavelets for efficient similarity search of time-series: with and without time warping," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 15, no. 3, pp. 686–705, 2003.

[23] D. Li, T. F. Bissyande, S. Kubler, J. Klein, and Y. Le Traon, "Profiling Household Appliance Electricity Usage with N-Gram Language Modeling," in *The 2016 IEEE International Conference on Industrial Technology (ICIT 2016)*. Taipei: IEEE, 2016, pp. 604–609.

[24] D. Li, L. Li, T. F. Bissyandé, J. Klein, and Y. Le Traon, "DSCo: A Language Modeling Approach for Time Series Classification," in *Machine Learning and Data Mining in Pattern Recognition: 12th International Conference, MLDM 2016, New York, NY, USA*, P. Perner, Ed. Springer International Publishing, July 2016, pp. 294–310.

[25] D. Li, T. F. Bissyandé, J. Klein, and Y. Le Traon, "DSCo-NG: A Practical Language Modeling Approach for Time Series Classification," in *Advances in Intelligent Data Analysis XV: 15th International Symposium, IDA 2015, Stockholm, Sweden.* Springer, October 2016, pp. 1–13.

[26] J. Serrà and J. L. Arcos, "An empirical evaluation of similarity measures for time series classification," *Knowledge-Based Systems*, vol. 67, pp. 305–314, 2014.

[27] S. Salvador and P. Chan, "Toward accurate dynamic time warping in linear time and space," *Intelligent Data Analysis*, vol. 11, no. 5, pp. 561–580, 2007.

[28] Y. Chen, E. Keogh, B. Hu, N. Begum, A. Bagnall, A. Mueen, and G. Batista, "The ucr time series classification archive," July 2015, www.cs.ucr.edu/~eamonn/time_series_data/.

[29] S. Varrette, P. Bouvry, H. Cartiaux, and F. Georgatos, "Management of an academic hpc cluster: The ul experience," in *Proc. of the 2014 Intl. Conf. on High Performance Computing & Simulation (HPCS 2014)*. Bologna, Italy: IEEE, July 2014, pp. 959–967.

[30] P. Nemenyi, "Distribution-free multiple comparisons," in *Biometrics*, vol. 18, no. 2, 1962, p. 263.