

Lecture Notes 5: Multiresolution Analysis

1 Frames

A frame is a generalization of an orthonormal basis. The inner products between the vectors in a frame and an arbitrary vector preserve the inner-product norm of the vector.

Definition 1.1 (Frame). *Let \mathcal{V} be an inner-product space. A frame of \mathcal{V} is a set of vectors $\mathcal{F} := \{\vec{v}_1, \vec{v}_2, \dots\}$ such that for every $\vec{x} \in \mathcal{V}$*

$$c_L \|\vec{x}\|_{\langle \cdot, \cdot \rangle}^2 \leq \sum_{\vec{v} \in \mathcal{F}} |\langle \vec{x}, \vec{v} \rangle|^2 \leq c_U \|\vec{x}\|_{\langle \cdot, \cdot \rangle}^2, \quad (1)$$

for fixed positive constants $c_U \geq c_L \geq 0$. The frame is a tight frame if $c_L = c_U$.

A direct consequence of the definition is that frames span the ambient space.

Lemma 1.2 (Frames span the whole space). *Any frame $\mathcal{F} := \{\vec{v}_1, \vec{v}_2, \dots\}$ of a vector space \mathcal{V} spans \mathcal{V} .*

Proof. Assume that there exists a vector \vec{y} that does not belong to the span, then $\mathcal{P}_{\text{span}(\vec{v}_1, \vec{v}_2, \dots)^\perp} \vec{y}$ is nonzero and orthogonal to all the vectors in the frame and cannot satisfy (1). \square

Orthonormal bases are examples of frames. They are frames that contain a minimum number of vectors.

Lemma 1.3 (Orthonormal bases are tight frames). *Any orthonormal basis $\mathcal{B} := \{\vec{b}_1, \vec{b}_2, \dots\}$ of a vector space \mathcal{V} is a tight frame.*

Proof. For any vector $\vec{x} \in \mathcal{V}$, by the Pythagorean theorem

$$\|\vec{x}\|_{\langle \cdot, \cdot \rangle}^2 = \left\| \sum_{\vec{b} \in \mathcal{B}} \langle \vec{x}, \vec{b} \rangle \vec{b} \right\|_{\langle \cdot, \cdot \rangle}^2 \quad (2)$$

$$= \sum_{\vec{b} \in \mathcal{B}} \left| \langle \vec{x}, \vec{b} \rangle \right|^2 \left\| \vec{b} \right\|_{\langle \cdot, \cdot \rangle}^2 \quad (3)$$

$$= \sum_{\vec{b} \in \mathcal{B}} \left| \langle \vec{x}, \vec{b} \rangle \right|^2. \quad (4)$$

\square

The operator that maps vectors to their frame coefficients is called the analysis operator of the frame.

Definition 1.4 (Analysis operator). *The analysis operator Φ of a frame maps a vector to its coefficients in the frame representation*

$$\Phi(\vec{x})[k] = \langle \vec{x}, \vec{v}_k \rangle. \quad (5)$$

For any finite frame $\{\vec{v}_1, \vec{v}_2, \dots, \vec{v}_m\}$ of \mathbb{C}^n the analysis operator corresponds to the matrix

$$F := \begin{bmatrix} \vec{v}_1^* \\ \vec{v}_2^* \\ \vdots \\ \vec{v}_m^* \end{bmatrix}. \quad (6)$$

In finite-dimensional spaces, any full rank square or tall matrix can be interpreted as the analysis operator of a frame.

Lemma 1.5 (Frames in finite-dimensional spaces). *A set of vectors $\{\vec{v}_1, \vec{v}_2, \dots, \vec{v}_m\}$ form a frame of \mathbb{C}^n if and only the matrix F defined by equation (6) is full rank. In that case,*

$$c_U = \sigma_1^2, \quad (7)$$

$$c_L = \sigma_n^2, \quad (8)$$

where σ_1 is the largest singular value of F and σ_n is the smallest.

Proof. By Theorem 2.7 in Lecture Notes 2, for any vector $\vec{x} \in \mathbb{C}^n$

$$\sigma_n^2 \leq \|F\vec{x}\|_2^2 = \sum_{j=1}^m \langle \vec{x}, \vec{v}_j \rangle^2 \leq \sigma_1^2. \quad (9)$$

□

To recover a vector from its frame coefficients, we need to invert the action of the analysis operator. In finite dimensions this can be achieved using the pseudoinverse of the corresponding matrix.

Lemma 1.6 (Pseudoinverse). *If an $n \times m$ tall matrix A , $m \geq n$, is full rank, then its pseudoinverse*

$$A^\dagger := (A^* A)^{-1} A^* \quad (10)$$

is well defined, is a left inverse of A

$$A^\dagger A = I \quad (11)$$

and equals

$$A^\dagger = V S^{-1} U^*, \quad (12)$$

where $A = USV^*$ is the SVD of A .

Proof.

$$A^\dagger := (A^* A)^{-1} A^* \quad (13)$$

$$= (V S U^* U S V^* A)^{-1} V S U^* \quad (14)$$

$$= (V S^2 V^*)^{-1} V S U^* \quad (15)$$

$$= V S^{-2} V^* V S U^* \quad (16)$$

$$= V S^{-1} U, \quad (17)$$

where S^{-2} and S^{-1} are diagonal matrices containing σ_j^{-2} and σ_j^{-1} in the j th entry of the diagonal, where σ_j denotes the j th singular value of A . These matrices are well defined as long as all the singular values are nonzero, or equivalently A is full rank. In that case,

$$A^\dagger A = V S^{-1} U V^* U S V^* \quad (18)$$

$$= I. \quad (19)$$

□

Corollary 1.7 (Recovering the signal). *Let \vec{c} be the representation of a vector \vec{x} in terms of a frame $\{\vec{v}_1, \vec{v}_2, \dots, \vec{v}_m\}$ of \mathbb{C}^n . Then applying the pseudoinverse of F recovers the signal from the coefficients*

$$\vec{x} = F^\dagger \vec{c}. \quad (20)$$

2 Short-time Fourier transform

The motivation to consider frames instead of bases is that they often make it possible to build signal decompositions that are more flexible. An important example is the short-time Fourier transform (STFT). Frequency representations such as the Fourier series and the DFT provide global information about the fluctuations of a signal, but they do not capture *local* information. However, the spectrum of speech, music and other sound signals changes continuously with time. The STFT is designed to describe these localized fluctuations. It consists of computing the frequency representation of a time segment of the signal, extracted through multiplication with a window.

Definition 2.1 (Short-time Fourier transform). *The short-time Fourier transform (STFT) of a function $f \in \mathcal{L}_2[-1/2, 1/2]$ is defined as*

$$\text{STFT}\{f\}(k, \tau) := \int_{-1/2}^{1/2} f(t) \overline{w(t - \tau)} e^{-i2\pi kt} dt, \quad (21)$$

where $w \in \mathcal{L}_2[-1/2, 1/2]$ is a window function. In words, it is equal to the Fourier series coefficients of the pointwise product between f and a shifted window $w_{[l]}$.

The STFT coefficients are equal to the inner product between the signal and vectors of the form $v_{k,\tau}(t) := w(t - \tau) e^{i2\pi kt}$, which corresponds to the window function w shifted by τ in time and by

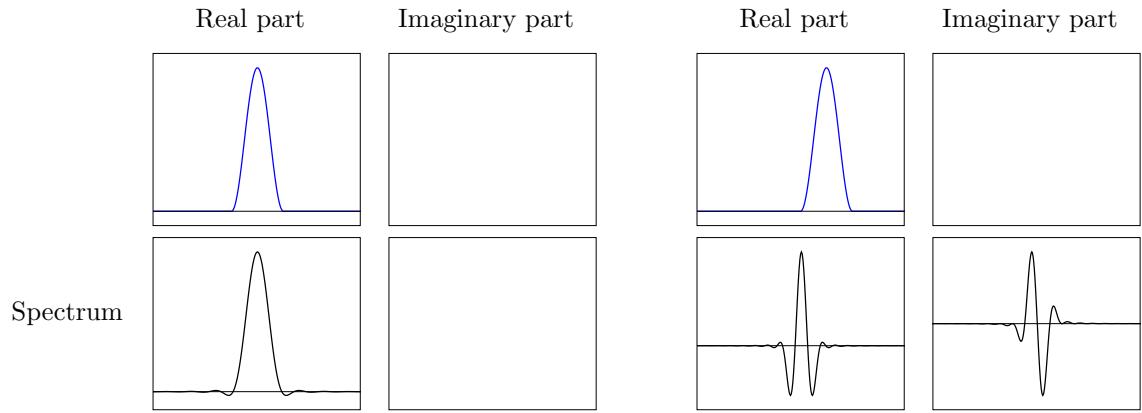
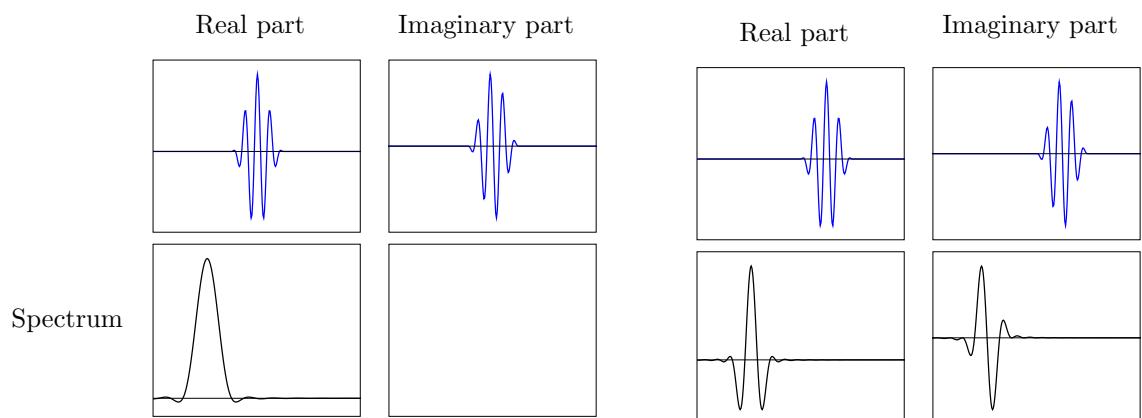
$\tau = 0, k = 0$ $\tau = 1/32, k = 0$  $\tau = 0, k = 64$ $\tau = 1/32, k = 64$ 

Figure 1: Examples of STFT frame vectors along with their Fourier representation.

k in frequency. As long as the shifts are chosen so that the windows overlap, the STFT coefficients form a frame. Figure 1 shows some of these frame vectors.

The discrete version of the short-time Fourier transform acts upon finite-dimensional vectors and is usually also known as STFT.

Definition 2.2 (Discrete short-time Fourier transform). *The STFT of a vector $\vec{x} \in \mathbb{C}^n$ is defined as*

$$\text{STFT } \{f\}(k, l) := \left\langle \vec{x} \circ \vec{w}_{[l]}, \vec{h}_k \right\rangle, \quad (22)$$

where \vec{w} is a window vector and $\vec{h}_k^{[n]}$ is the discrete complex sinusoidal vector from Definition 1.5 in Lecture Notes 4.

As in the continuous case, if the shifts overlap sufficiently, then this transformation is a frame in a finite-dimensional space. This means that there is a tall matrix that represents the analysis operator, and that we can invert it with the pseudoinverse by Lemma 1.6. However this would be very inefficient computationally! The STFT operator is usually applied and inverted using fast algorithms based on the FFT.

The simplest window function that we can use is a rectangular function, i.e. just selecting intervals of coefficients. Unfortunately, this introduces an artificial discontinuity at the ends of the interval. Mathematically, multiplying the coefficients by the rectangular function is equivalent to convolving with a Dirichlet kernel in the spectral domain, which becomes apparent when we compute the Fourier coefficients of the windowed data, as shown in Figure 2. In contrast, Gaussian-like windows that taper off at the ends smooth the borders of the windowed signal and avoid the high-frequency artifacts introduced by the side lobes of the Dirichlet kernel.

The STFT is an important tool for sound processing. Variations in the spectral components of signals are visualized using the *spectrogram*, which is equal to the logarithm of the magnitude of the STFT coefficients. Figure 4 shows the spectrogram of a real speech signal. The time and frequency representation of the same signal are shown in Figure 3. In contrast to these representations, the spectrogram reveals how the frequency components of the speech signal vary over time. The resolution at which we track these variations depends on the width of the window chosen to compute the spectrogram. Shorter windows provide higher temporal resolution (we can track quicker changes), but are not able to detect lower-frequency components whose periods are longer than the chosen window. This motivates using windows of different lengths to extract information at multiple resolutions. The next section discusses signal representations designed to achieve this.

3 Wavelets

3.1 Definition

Wavelets are designed to capture signal structure at different scales. This is achieved with an analysis operator that contains scaled copies of a fixed function called a wavelet.

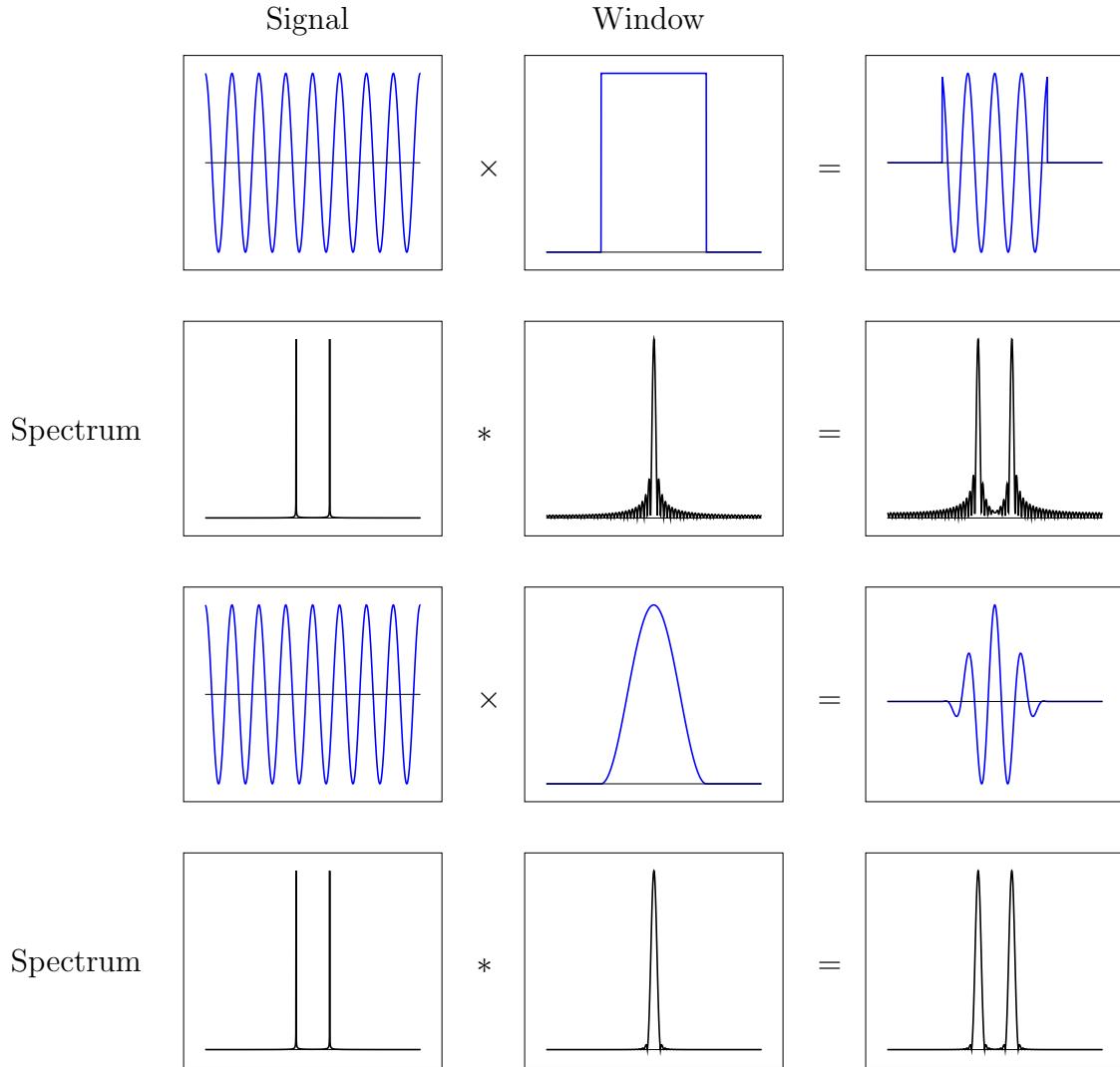


Figure 2: The spectrum of a time segment may contain spurious high-frequency content produced by the sudden transition at the ends of the segment. In the frequency domain, the spectrum is being convolved by a sinc function, which has a very heavy tail. Multiplying the signal by a localized window that has a faster decay in the frequency domain alleviates the problem.

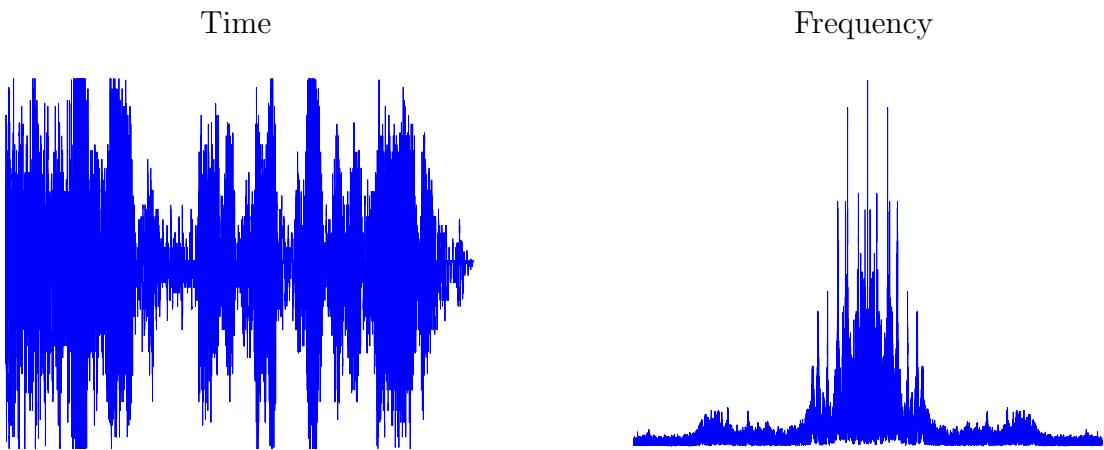


Figure 3: Time and frequency representation of a speech signal.

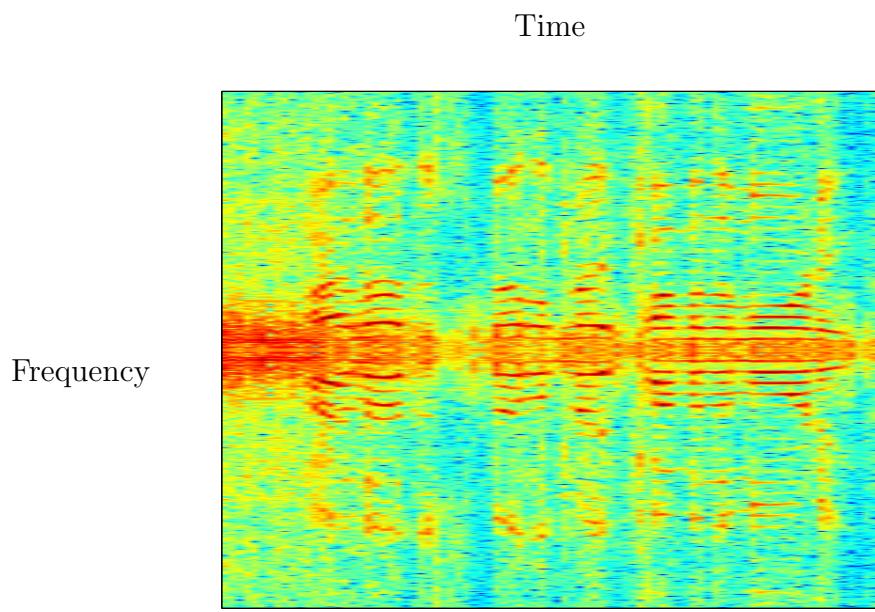


Figure 4: Spectrogram (log magnitude of STFT coefficients) of the speech signal in Figure 3.

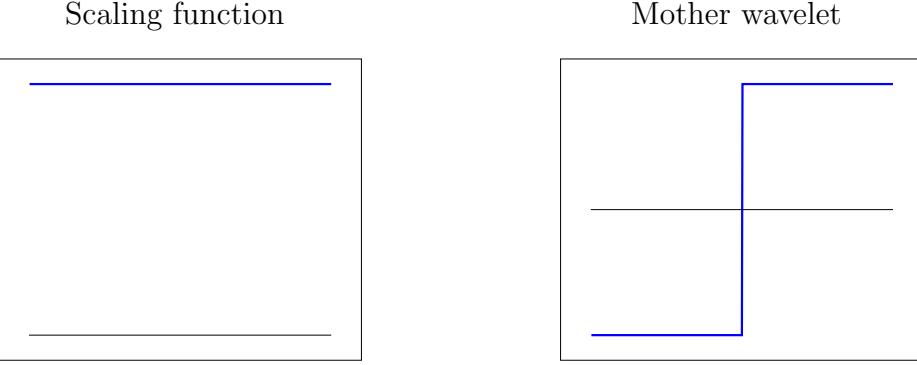


Figure 5: Scaling and wavelet function of the Haar wavelet transform.

Definition 3.1 (Wavelet transform). *The wavelet transform of a function $f \in \mathcal{L}_2$ depends on a choice of wavelet (or mother wavelet) $\psi \in \mathcal{L}_2$ and scaling function $\phi \in \mathcal{L}_2$ (or father wavelet). The scaling coefficients are obtained through an inner product with shifted copies of ϕ*

$$W_\phi \{f\}(\tau) := \frac{1}{\sqrt{s}} \int f(t) \overline{\phi(t-\tau)} dt \quad (23)$$

whereas the wavelet coefficients are obtained through an inner product with dilated, shifted copies of ψ

$$W_\psi \{f\}(s, \tau) := \frac{1}{\sqrt{s}} \int f(t) \overline{\psi\left(\frac{t-\tau}{s}\right)} dt. \quad (24)$$

Intuitively, $W\{f\}(s, \tau)$ captures the information at scale s and location τ . The scaling function can be interpreted as a low-pass filter that extracts the global features that are not captured by the wavelet coefficients. The Haar wavelet is an example of a wavelet. Figure 5 shows its scaling and wavelet functions.

Definition 3.2 (Haar wavelet). *The scaling function for the Haar wavelet is a rectangular function*

$$\phi(t) := 1, \quad -\frac{1}{2} \leq t \leq \frac{1}{2}. \quad (25)$$

The Haar wavelet is of the form

$$\psi(t) := \begin{cases} -1, & -\frac{1}{2} \leq t \leq 0, \\ 1, & 0 \leq t \leq \frac{1}{2}. \end{cases} \quad (26)$$

The discrete wavelet transform acts upon finite-dimensional vectors.

Definition 3.3 (Discrete wavelet transform). *The wavelet transform of a function $f \in \mathbb{C}^n$ depends on a choice of wavelet (or mother wavelet) $\vec{\psi} \in \mathbb{C}^n$ and scaling vector $\vec{\phi} \in \mathbb{C}^n$ (or father wavelet). The scaling coefficients are obtained through an inner product with shifted copies of $\vec{\phi}$*

$$W_{\vec{\phi}} \{f\}(l) := \langle \vec{x}, \vec{\phi}_{[l]} \rangle, \quad (27)$$

whereas the wavelet coefficients are obtained through an inner product with scaled, shifted copies of $\vec{\psi}$

$$W_{\vec{\psi}}\{f\}(s, l) := \left\langle \vec{x}, \vec{\psi}_{[s,l]} \right\rangle, \quad (28)$$

where

$$\vec{\psi}_{[s,l]}[j] := \vec{\psi} \left[\frac{j-l}{s} \right]. \quad (29)$$

The discrete Haar wavelet is the discrete counterpart of the Haar wavelet.

Definition 3.4 (Discrete Haar wavelet). *The scaling function for the Haar wavelet is a rectangular function*

$$\vec{\phi}[j] := 1, \quad 1 \leq j \leq n. \quad (30)$$

The Haar wavelet is of the form

$$\vec{\psi}[j] := \begin{cases} -1, & j = \frac{n}{2}, \\ 1, & j = \frac{n}{2} + 1, \\ 0, & \text{otherwise.} \end{cases} \quad (31)$$

3.2 Multiresolution decomposition

The discrete Haar wavelet and its corresponding scaling vector can be used to construct a basis of \mathbb{C}^n .

Definition 3.5 (Haar wavelet basis). *Let $n := 2^K$ for some integer K . We fix a single scaling vector*

$$\vec{\phi}[j] := \frac{1}{\sqrt{n}}, \quad 1 \leq j \leq n. \quad (32)$$

We fix $K+1$ scales equal to $2^0, 2^1, \dots, 2^K$. The wavelets at the finest scale 2^0 are given by

$$\vec{\psi}[j] := \begin{cases} -\frac{1}{\sqrt{2}}, & j = 1, \\ \frac{1}{\sqrt{2}}, & j = 2, \\ 0, & j > 2, \end{cases} \quad (33)$$

and copies of $\vec{\psi}$ shifted by 2, so that they do not overlap. The wavelets at scale 2^k , $1 \leq k \leq K$ are copies of $\vec{\psi}$ dilated by 2^k , multiplied by a factor of $1/\sqrt{2^k}$ (their ℓ_2 norm equals one) and shifted by multiples of 2^{k+1} (so the basis vectors at each scale do not overlap).

The Haar wavelet basis contains n unit-norm vector that are all orthogonal, so they form an orthonormal basis of \mathbb{C}^n . Figure 6 shows the basis vectors for $n = 8$. Figure 7 shows the coefficients of an electrocardiogram signal in the basis.

Wavelet bases can be interpreted as a multiresolution representation, where the coefficients corresponding to different dilations of the wavelet capture information at the corresponding scales. This is made more precise in the following definition.

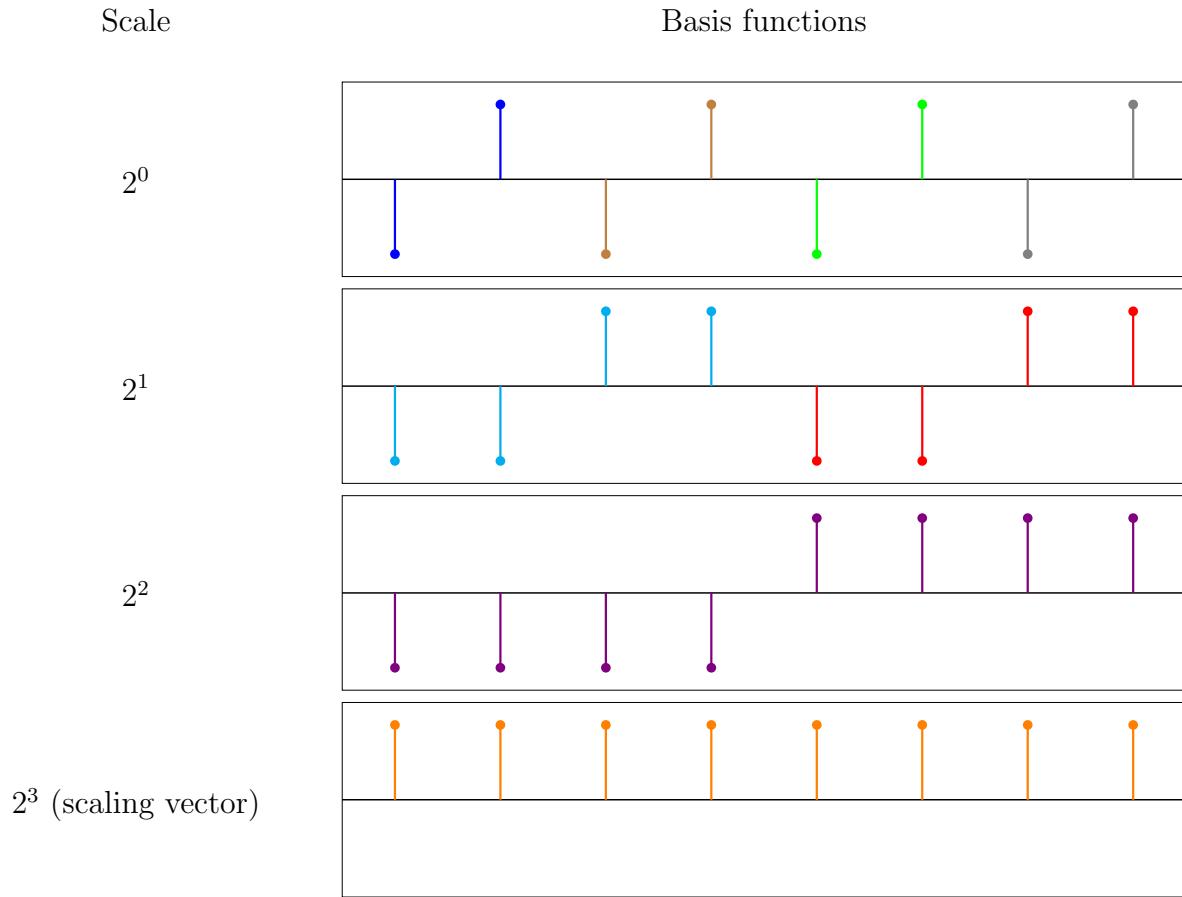


Figure 6: Basis functions in the Haar wavelet basis for \mathbb{C}^8 .

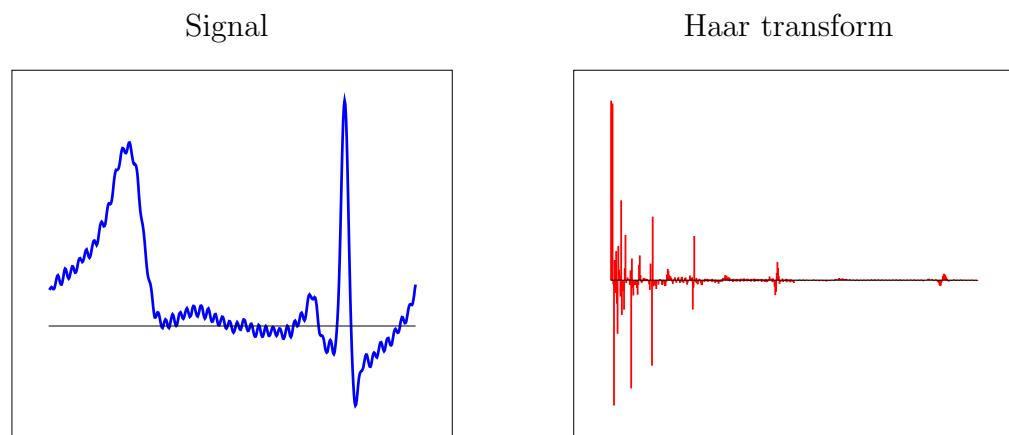


Figure 7: Electrocardiogram signal (left) and its corresponding Haar wavelet coefficients (right).

Definition 3.6 (Multiresolution wavelet decomposition). *Let $n := 2^K$ for some integer K . Given a scaling vector $\vec{\phi} \in \mathbb{C}^n$ and a wavelet $\vec{\psi} \in \mathbb{C}^n$, a multiresolution decomposition of \mathbb{C}^n is a sequence of subspaces $\mathcal{V}_0, \mathcal{V}_1, \dots, \mathcal{V}_K$ where:*

- \mathcal{V}_K is spanned by the scaling vector $\vec{\phi}$.
- $\mathcal{V}_k := \mathcal{W}_k \oplus \mathcal{V}_{k+1}$ where \mathcal{W}_k is the span of ψ dilated by 2^k and shifted by multiples of 2^{k+1} .

For any vector $\vec{x} \in \mathbb{C}^n$, $\mathcal{P}_{\mathcal{V}_k} \vec{x}$ is the approximation of \vec{x} at scale 2^k .

To be a valid multiresolution decomposition, the subspaces must satisfy the following properties:

- $\mathcal{V}_0 = \mathbb{C}^n$, the approximation at scale 2^0 is perfect.
- \mathcal{V}_k is invariant to translations of scale 2^k for $0 \leq k \leq K$. If $\vec{x} \in \mathcal{V}_k$ then

$$\vec{x}_{[2^k l]} \in \mathcal{V}_k \quad \text{for all } l \in \mathbb{Z}, \quad (34)$$

where the shifts are circular.

- Dilating vectors in \mathcal{V}_j by 2 yields vectors in \mathcal{V}_{j+1} . Let $\vec{x} \in \mathcal{V}_j$ be nonzero only between 1 and $n/2$, the dilated vector \vec{y} defined by

$$\vec{y}[j] = \vec{x}[\lceil j/2 \rceil] \quad (35)$$

belongs to \mathcal{V}_{j+1} .

By construction, the Haar wavelet basis in Definition 3.5 provides a multiresolution decomposition of \mathbb{C}^n . In Figure 8 the decomposition is applied to obtain approximations of an electrocardiogram signal at different scales. Many other wavelet bases apart from the Haar yield multiresolution decompositions: Meyer, Daubechies, Battle-Lemarie, ... We refer the interested reader to Chapter 7 in [?] for a detailed and rigorous description of the construction of orthonormal wavelet bases.

3.3 Multidimensional wavelet decompositions

Two-dimensional wavelets can be obtained by taking outer products of one-dimensional wavelets, as in the case of the two-dimensional discrete Fourier transform. 2D wavelets are of the form,

$$\xi_{[s_1, s_2, k_1, k_2]}^{2D} := \xi_{[s_1, k_1]}^{1D} (\xi_{[s_2, k_2]}^{1D})^*, \quad (36)$$

where ξ can refer to both 1D scaling and wavelet functions. We consider shifts k_1, k_2 in two dimensions and a two-dimensional scaling s_1, s_2 . The corresponding two-dimensional transform allows to obtain multiscale representations of images. An example is shown in Figure 10. The coefficients are grouped by their scale (which is decreasing as we move down and to the right) and arranged in two dimensions, according to the location of the corresponding shifted wavelet with respect to the image. Figure 9 shows the vectors in a 2D Haar wavelet basis.

Designing multidimensional transforms that are more effective at providing sparse representations for images has been a vibrant research subject for many years. Some of these extensions include the steerable pyramid, ridgelets, curvelets, and bandlets. We refer to Section 9.3 in [?] for more details.

Scale (2^k)

$\mathcal{P}_{\mathcal{W}_k} \vec{x}$

$\mathcal{P}_{\mathcal{V}_k} \vec{x}$

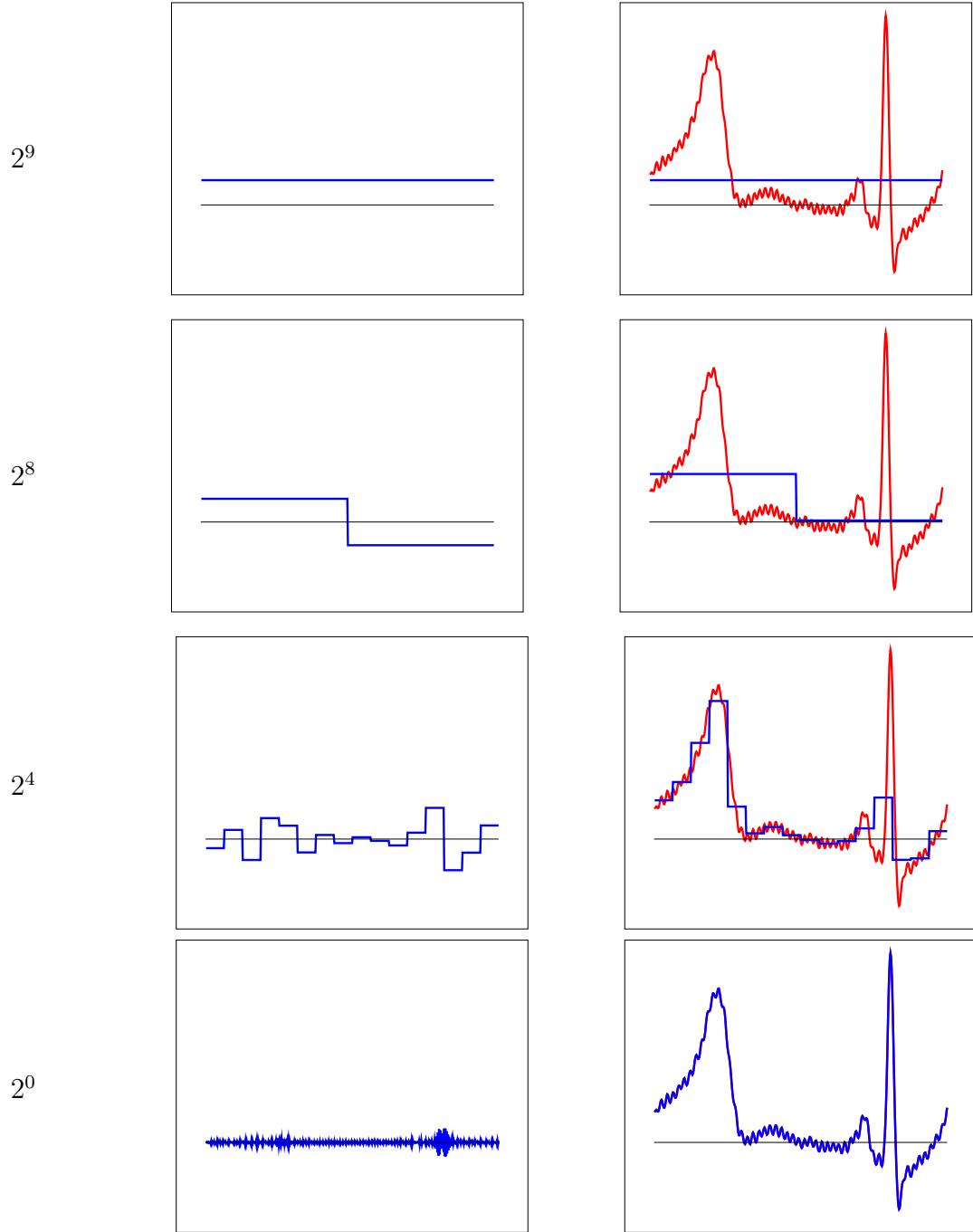


Figure 8: Multiresolution decomposition of the electrocardiogram signal in Figure 7. On the left, the projection of the signal onto \mathcal{W}_k extracts information at scale 2^k . On the right, projection onto \mathcal{V}_k yields an approximation of the signal at scale 2^k .

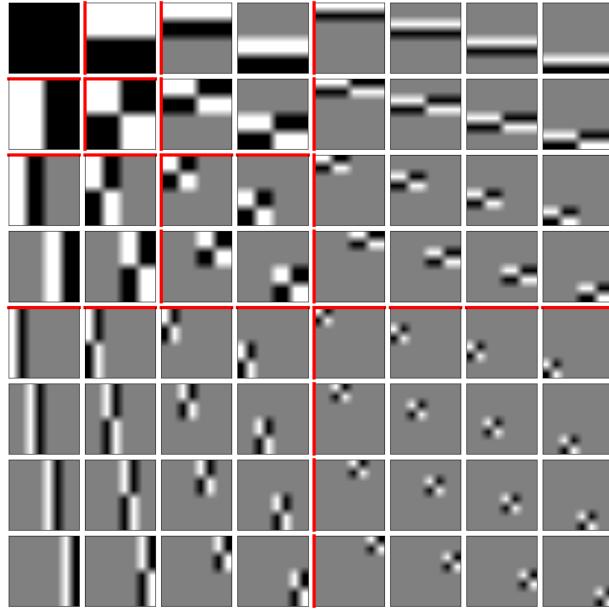


Figure 9: Basis vectors of the 2D Haar wavelet transform.

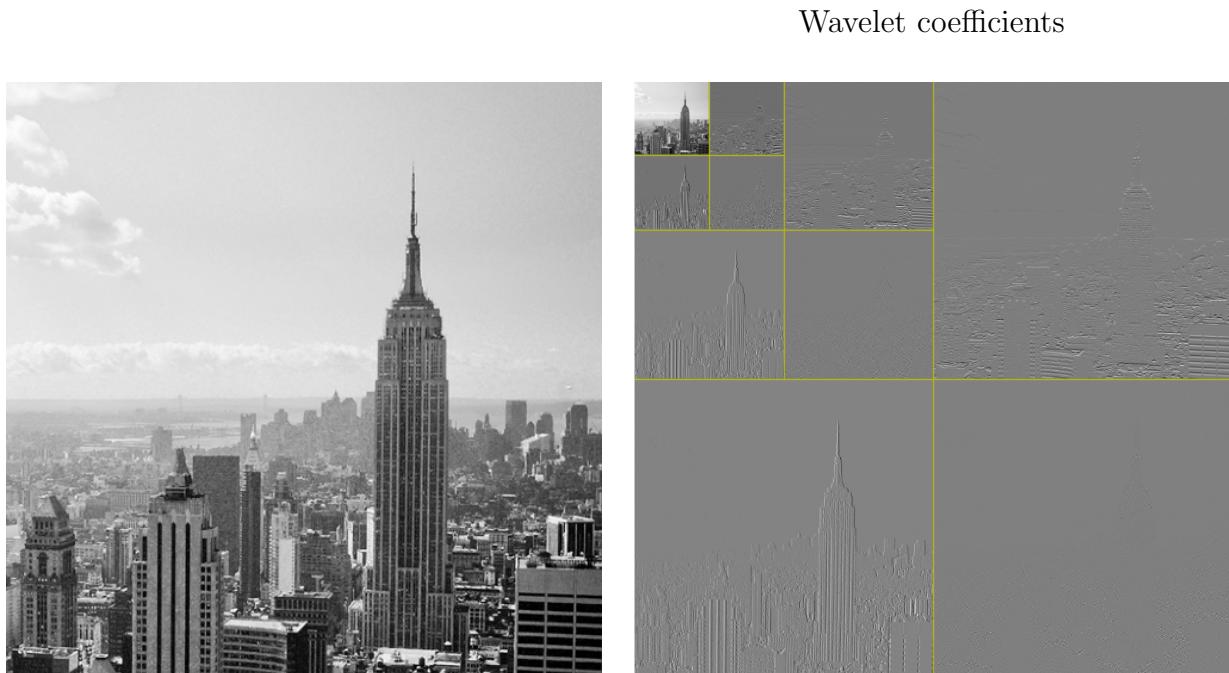
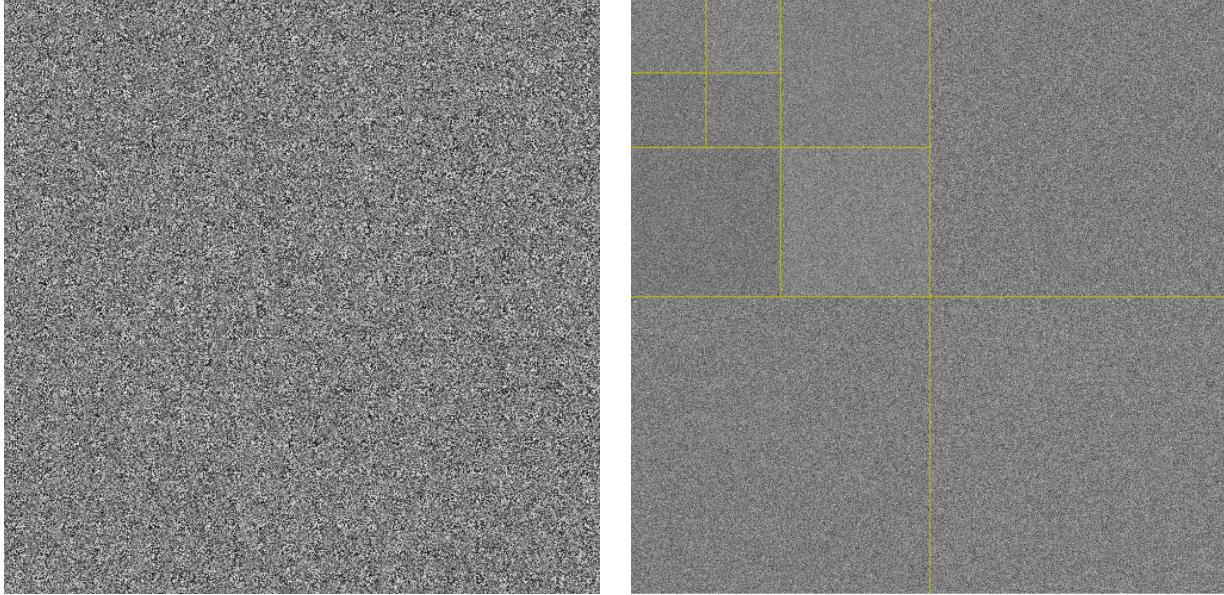


Figure 10: An image (left) and its coefficients in a 2D Haar wavelet basis (right). The coefficients are arranged so that the scaling coefficients are on the top left and coefficients corresponding to increasingly fine scales are situated below and to the right.

Wavelet coefficients

**Figure 11:** Gaussian iid noise (left) and its Haar wavelet coefficients (left).

4 Denoising via thresholding

The STFT and wavelet transforms often yield **sparse** signal representations, meaning that many coefficients are equal to zero. In the case of the STFT, this occurs when only a few spectral components are active at a particular time, which is typical of speech or music signals (see Figure 4). In the case of wavelets, sparsity results from the fact that large regions of natural images (and many other signals) are smooth and mostly contain coarse-scale features, whereas most of the fine-scale features are confined to edges or regions with high-frequency textures.

In contrast, noisy perturbations usually have dense coefficients in any fixed frame or basis. As we establish in Lecture Notes 3, if \vec{z} is a Gaussian random vector with covariance matrix $\sigma^2 I$, for some fixed $\sigma^2 > 0$ then $F\vec{z}$ is a Gaussian random vector with covariance matrix FF^* . In particular, if F is a basis, then $F\vec{z}$ is iid Gaussian, which means that the magnitude of most entries is approximately equal to the standard deviation σ . Figure 11 shows the Haar wavelet coefficients of iid Gaussian noise. As expected, the coefficients are also noisy and dense in this representation.

Let us consider the problem of denoising measurements $\vec{y} \in \mathbb{C}^n$ of a signal $\vec{x} \in \mathbb{C}^n$ corrupted by additive noise $\vec{z} \in \mathbb{C}^n$

$$\vec{y} := \vec{x} + \vec{z}. \quad (37)$$

Under the assumptions that (1) $F\vec{x}$ is sparse representation where F is a certain frame or basis and (2) the entries of $F\vec{z}$ are small and dense, thresholding $F\vec{y}$ makes it possible to suppress the noise while preserving the signal. Figure 12 shows an example of two noisy images with different signal-to-noise ratios (SNR), defined as the ratio between the ℓ_2 norm of the signal and the noise. In the wavelet domain, the coefficients corresponding to the signal lie above a *sea* of noisy coefficients.

SNR

2.5



1



Wavelet coefficients

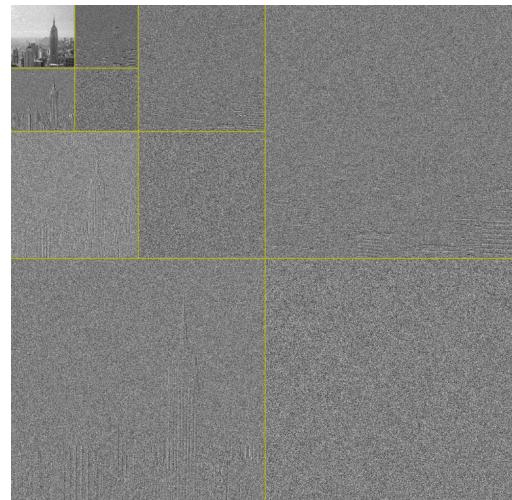
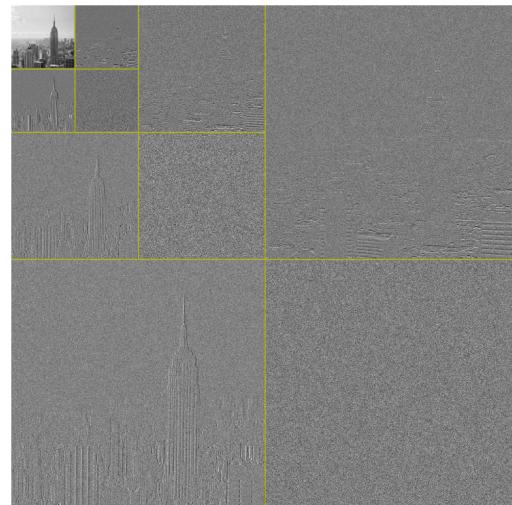


Figure 12: The two noisy images on the left are obtained by adding Gaussian noise to the image in Figure 10 to obtain an SNR of 2.5 (above) and 1 (below). The coefficients of the images in the 2D Haar basis are shown on the right.

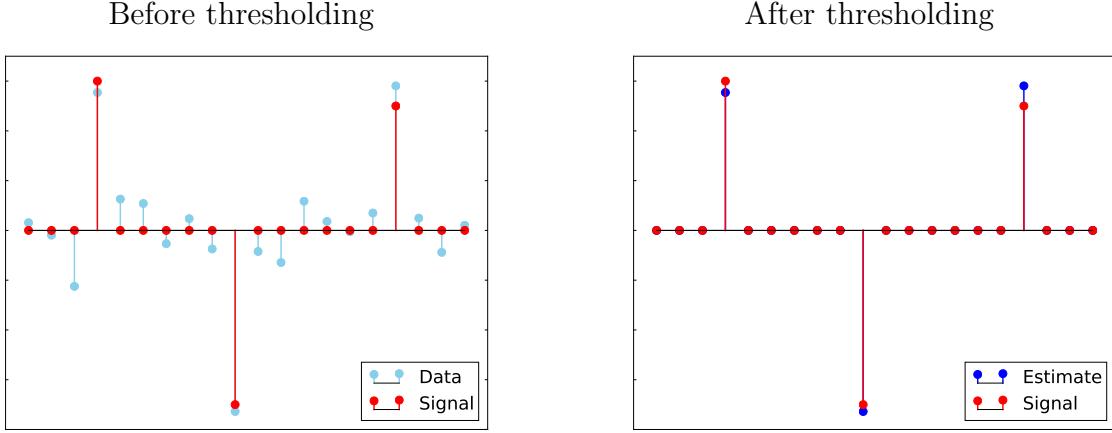


Figure 13: Denoising via hard thresholding.

To motivate threshold-based denoising, consider the case where \vec{x} itself is sparse and \vec{z} . In that case we can denoise by setting to zero the entries in \vec{y} that are below a certain value. Figure 13 illustrates this with a simple example. Most signals are not sparse, but in many cases we can design a linear transform that sparsifies them. We can then apply the same idea to the coefficients of the measurements in this representation.

Algorithm 4.1 (Denoising via hard thresholding). *Let \vec{y} follow the model in equation (37). To estimate the signal we:*

1. *Compute a decomposition $F\vec{y}$, where F is a frame or basis which sparsifies the signal \vec{x} .*
2. *Apply the hard-thresholding operator $\mathcal{H}_\eta : \mathbb{C}^n \rightarrow \mathbb{C}^n$ to $F\vec{y}$*

$$\mathcal{H}_\eta(\vec{v})[j] := \begin{cases} \vec{v}[j] & \text{if } |\vec{v}[j]| > \eta, \\ 0 & \text{otherwise,} \end{cases} \quad (38)$$

for $1 \leq j \leq n$, where η is adjusted according to the standard deviation of $F\vec{z}$. If F is a basis and \vec{z} is iid Gaussian with standard deviation σ , η should be set larger than σ .

3. *Compute the estimate by inverting the transform. If F is a basis, then*

$$\vec{x}_{\text{est}} := F^{-1}\mathcal{H}_\eta(F\vec{y}). \quad (39)$$

If F is a frame,

$$\vec{x}_{\text{est}} := F^\dagger \mathcal{H}_\eta(F\vec{y}), \quad (40)$$

where F^\dagger is the pseudoinverse of F (any other left inverse of F would also work).

Figure 14 shows the result of denoising a multisinusoidal signal by thresholding its Fourier coefficients. Figure 15 shows the result of denoising the images in Figure 12 by thresholding their 2D wavelet coefficients.

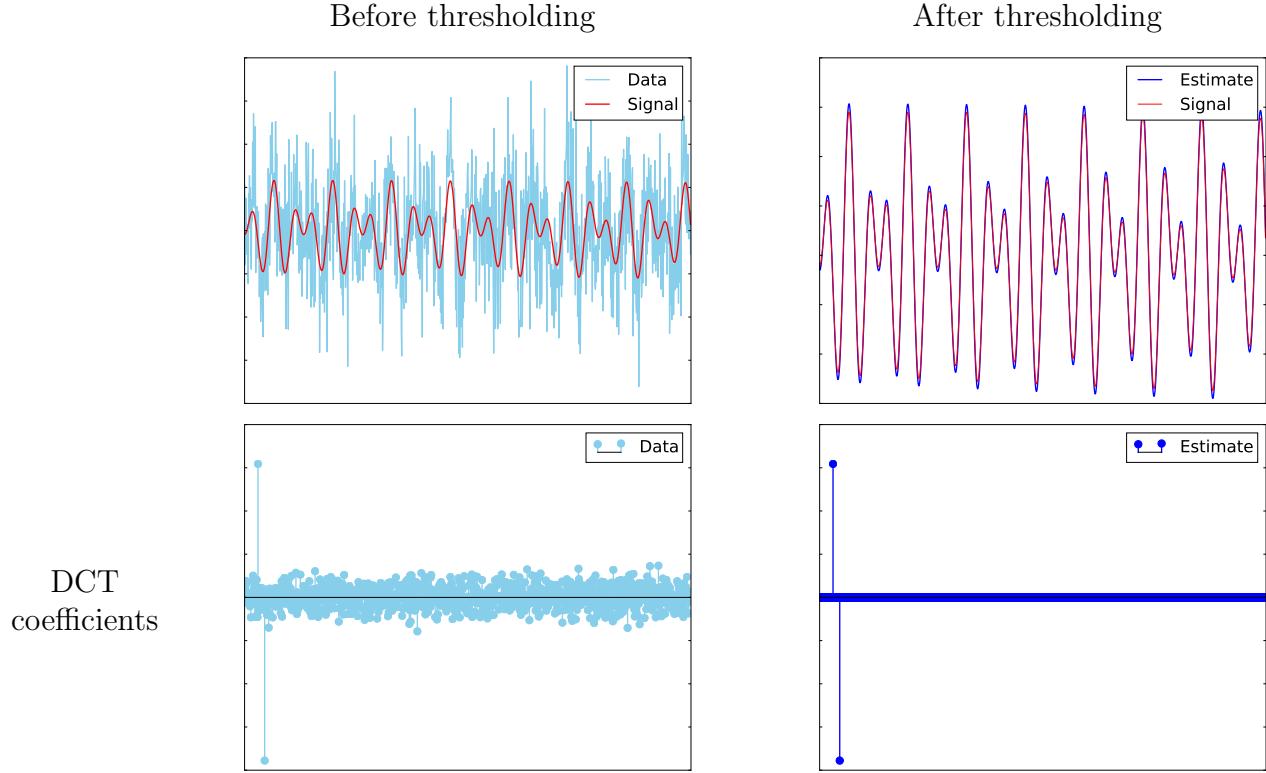


Figure 14: Denoising via hard thresholding in a Fourier basis.

When we apply transforms that capture localized details of signals, such as the wavelet transform or the STFT, sparse representations tend to be highly structured. For example, nonzero wavelet coefficients are often clustered around edges. This is apparent in Figure 10. The reason is that several localized atoms are needed to reproduce sharp variations, whereas a small number of coarse-scale atoms suffice to represent smooth areas of the image.

Thresholding-based denoising can be enhanced by taking into account the *group sparsity* of the signal of interest. If we have a reason to believe that nonzero coefficients in the signal tend to be close to each other, then we should threshold small isolated coefficients, but not similar coefficients that are in the vicinity of large coefficients and therefore may contain useful information. This can be achieved by applying block thresholding.

Algorithm 4.2 (Denoising via block thresholding). *Let \vec{y} follow the model in equation (37). To estimate the signal we:*

1. Compute a decomposition $F\vec{y}$, where F is a frame or basis which sparsifies the signal \vec{x} .
2. Partition the indices of $F\vec{y}$ into blocks $\mathcal{I}_1, \mathcal{I}_2, \dots, \mathcal{I}_k$.
3. Apply the hard-block-thresholding operator $\mathcal{B}_\eta : \mathbb{C}^n \rightarrow \mathbb{C}^n$ to $F\vec{y}$

$$\mathcal{B}_\eta(\vec{v})[j] := \begin{cases} \vec{v}[j] & \text{if } j \in \mathcal{I}_j \text{ such that } \|\vec{v}_{\mathcal{I}_j}\|_2 > \eta, \\ 0 & \text{otherwise,} \end{cases} \quad (41)$$

SNR

2.5



1



Wavelet coefficients

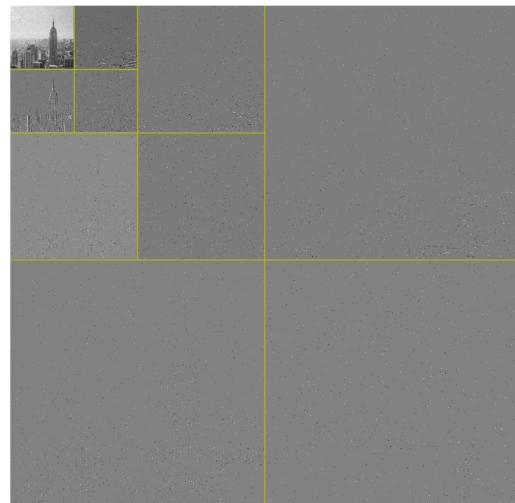
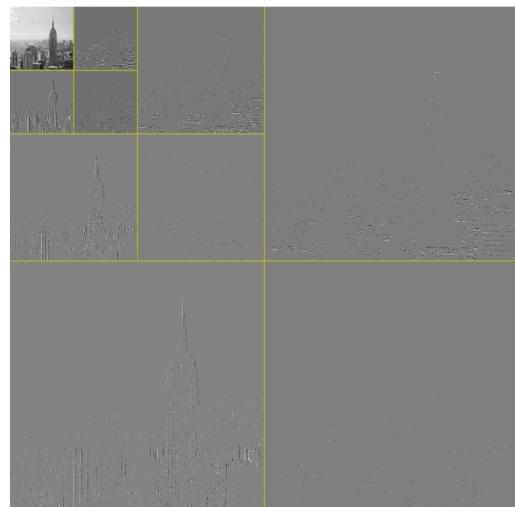


Figure 15: Thresholding-based denoising applied to the images in Figure 12.

SNR

2.5



1



Wavelet coefficients

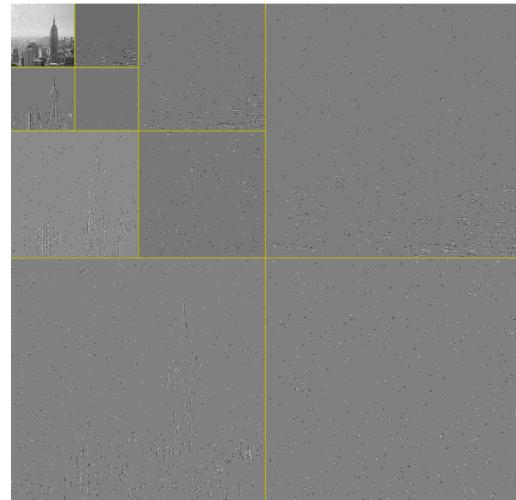
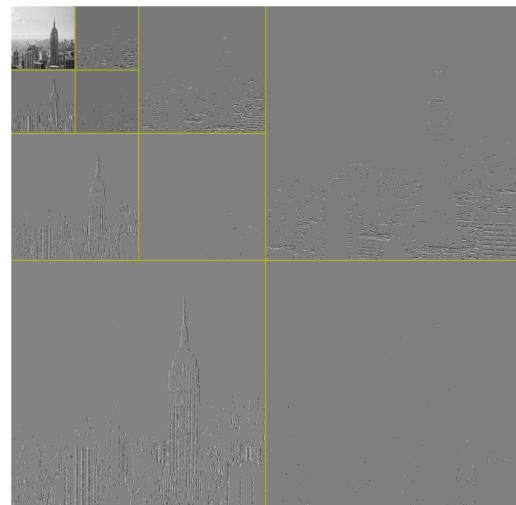


Figure 16: Block-thresholding-based denoising applied to the images in Figure 12.

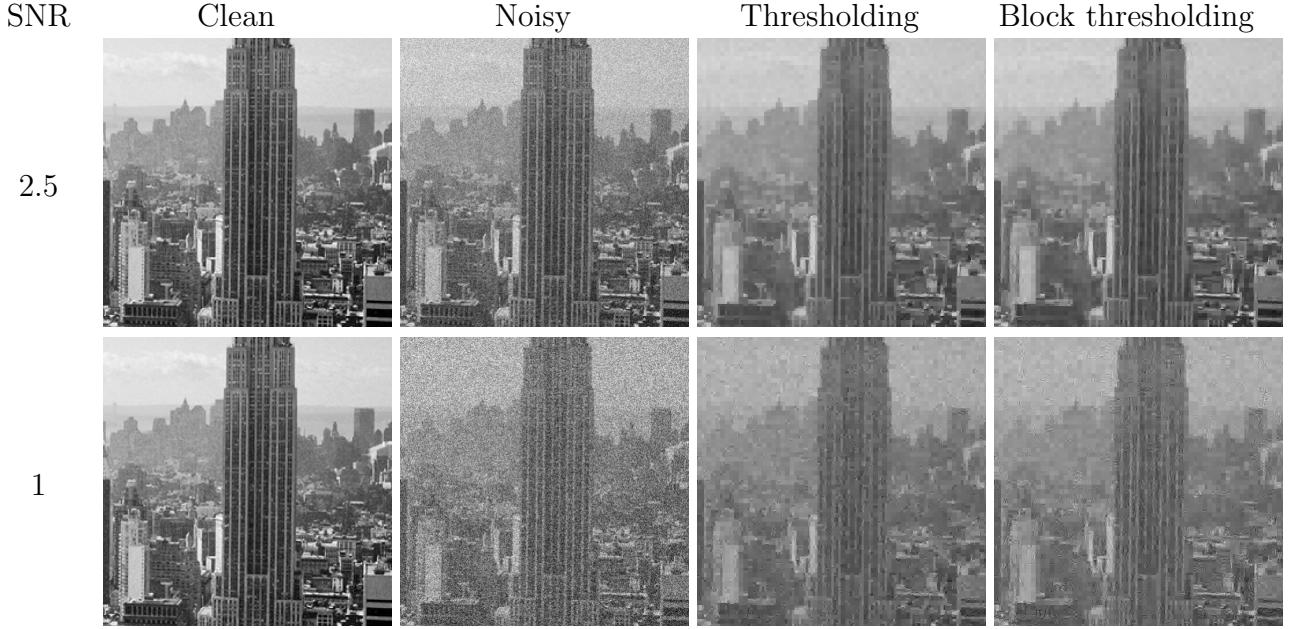


Figure 17: Comparison between thresholding and block-thresholding applied to denoise the images in Figure 12.

where η is adjusted according to the standard deviation of $F\vec{z}$. If F is a basis and \vec{z} is iid Gaussian with standard deviation σ , η should be set larger than $b\sigma$, where b is the number of indices in each block.

4. Compute the estimate by inverting the transform. If F is a basis, then

$$\vec{x}_{\text{est}} := F^{-1}\mathcal{B}_\eta(F\vec{y}). \quad (42)$$

If F is a frame,

$$\vec{x}_{\text{est}} := F^\dagger\mathcal{B}_\eta(F\vec{y}), \quad (43)$$

where F^\dagger is the pseudoinverse of F (any other left inverse of F would also work).

Figure 16 shows the result of denoising the images in Figure 12 by partitioning its 2D Haar coefficients in 4×4 blocks and applying block thresholding. As illustrated by Figure 17 block-thresholding recovers regular such as the vertical lines on the Empire State building more effectively.

We conclude this section with an application of thresholding-based denoising to speech.

Example 4.3 (Speech denoising). The recording shown in Figures 3 and 4 is a short snippet from the movie *Apocalypse Now* where one of the character talks over the noise of a helicopter. We denoise the data using the following methods (click on the links to hear the result):

- **Time thresholding:** The result, which is plotted in Figure 18, sounds terrible because the thresholding eliminates parts of the speech.

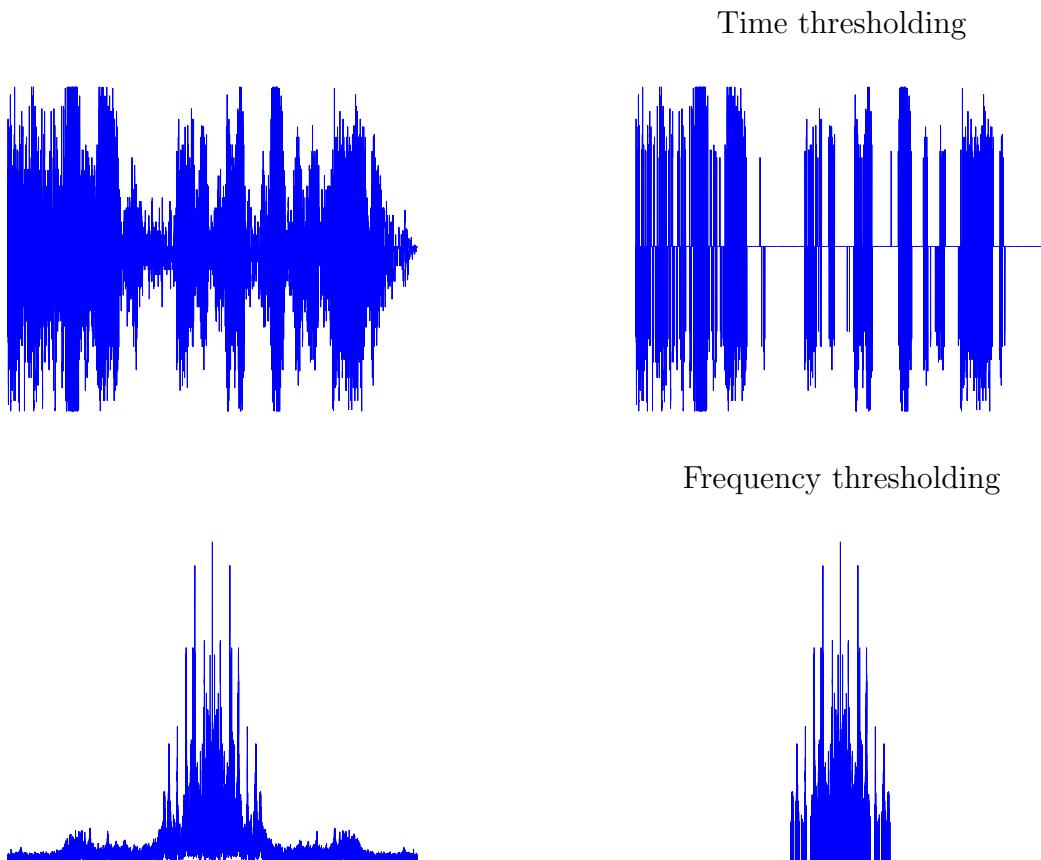


Figure 18: Time thresholding (top row) applied to the noisy data shown in Figure 3. The [result](#) sounds terrible because the thresholding eliminates parts of the speech. Below, frequency thresholding is applied to the same data. The [result](#) is very low pitch because the thresholding eliminates the high frequencies of both the speech and the noise.

- **Frequency thresholding:** The result has very low pitch because the thresholding eliminates the high frequencies of both the speech and the noise. The spectrum is shown in Figure ?? before and after thresholding.
- **STFT thresholding:** The result is significantly better but isolated STFT coefficients that are not discarded produce *musical noise* artifacts. The corresponding spectrogram is shown in Figure 19.
- **STFT block thresholding:** The result does not suffer from musical noise and retains some of the high-pitch speech. The corresponding spectrogram is shown in Figure 19.

The results are compared visually for a small time segment of the data in Figure 20. △

References

For more information on multiresolution approximations and time-frequency signal processing we refer to the excellent book [?] and references therein.

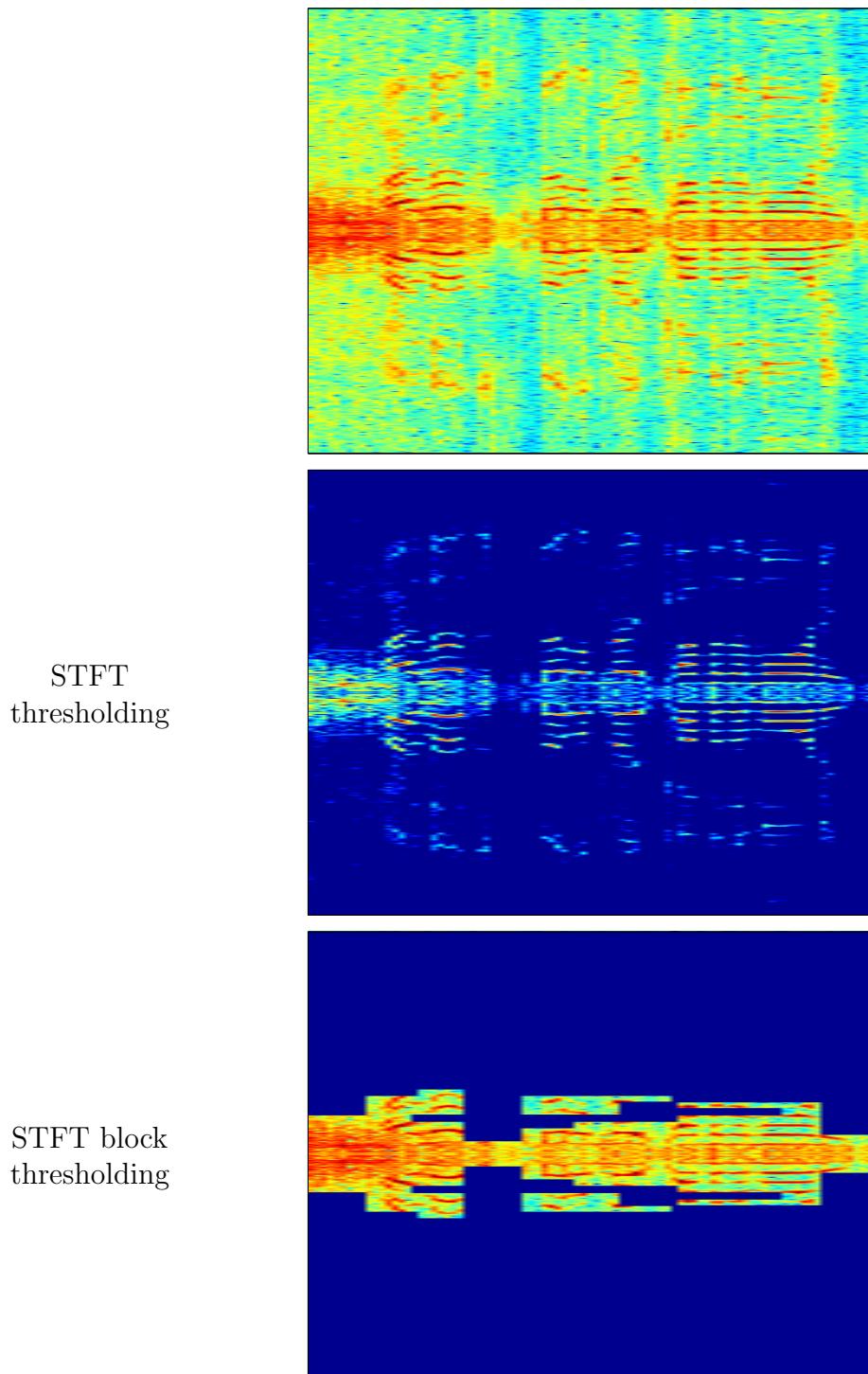


Figure 19: Spectrograms of the noisy signal (above) compared to the estimates obtained by simple thresholding (center) and block thresholding (bottom). The result of [simple thresholding](#) contains musical noise caused by particularly large STFT coefficients caused by the noise that were not thresholded. The result of [block thresholding](#) does not suffer from these artifacts.

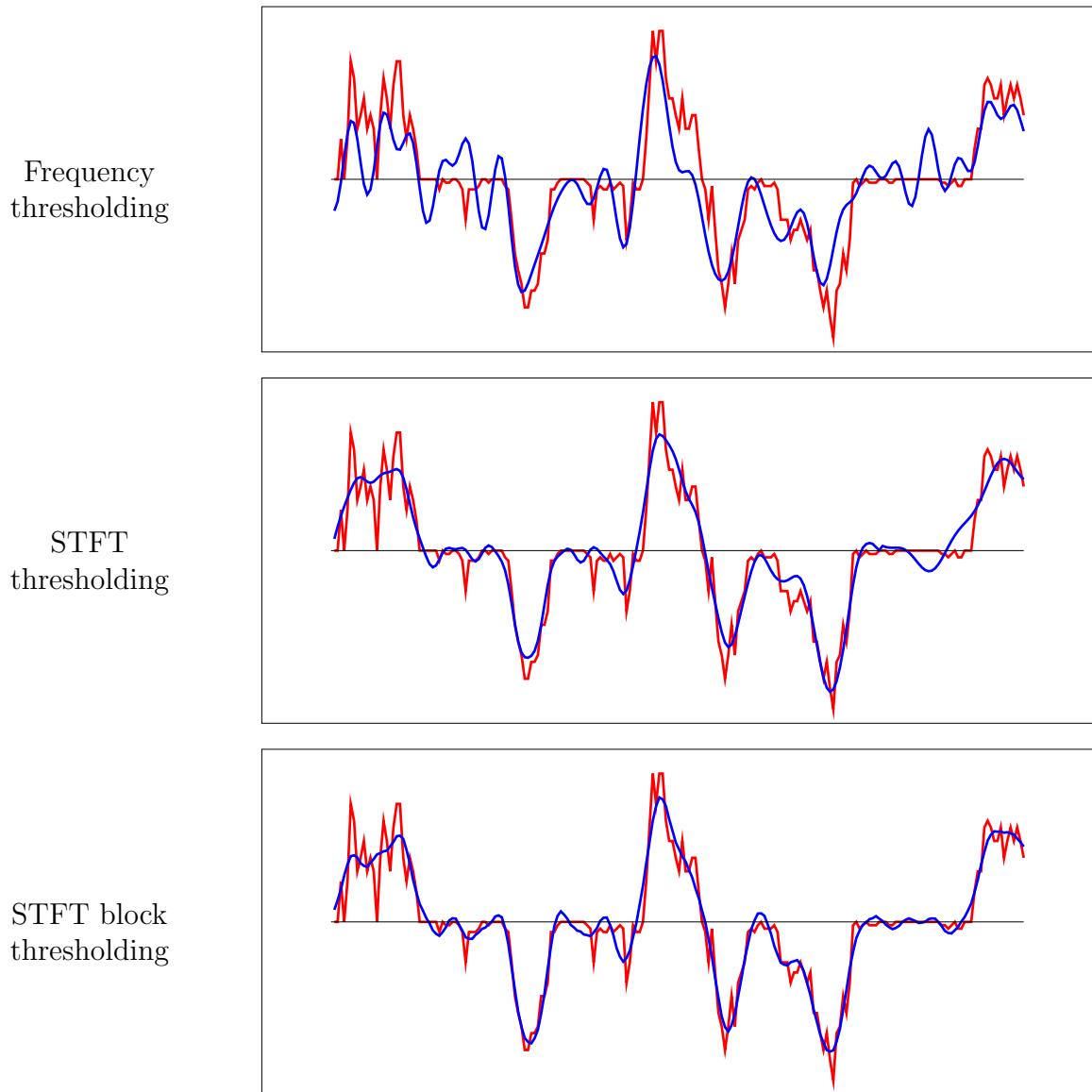


Figure 20: Comparison of the original noisy data (blue) with the denoised signal for the data shown in Figure 3. We compare frequency thresholding (above) and thresholding (center) and block thresholding (below) of STFT coefficients.