

Signal Representations Beyond Fourier

Fourier representations, including the Fourier series and the DFT, are a pillar of signal processing, but they have a fundamental limitation: Fourier coefficients capture global fluctuations of a signal, and hence do not provide information about events that are localized in time or space. In this chapter we study signal representations that address this issue. Section 1 describes the short-time Fourier transform, designed to capture frequency information that is localized in time. Section 2 introduces wavelet transforms, which decompose signals into components with different resolutions. As explained, in Section 3, these transformations can be used to generate sparse representations of audio and image signals, which are useful for tasks like denoising.

1 Time-frequency representations

1.1 Signal segmentation using windows

The Fourier series and the DFT reveal the periodic components of signals. However, they do not provide any information about the time structure of these periodic components. Consider the audio signal in Figure 1. Like most speech and music signals, the signal consists of periodic oscillations that *change over time*. These periodic components are present in its Fourier series coefficients, but it is impossible to discern at what times each of them is present or absent. An effective way of capturing such structure is to compute the Fourier coefficients of a signal after segmenting it into intervals. Segmentation can be achieved through multiplication with a window function, which is nonzero only over a specific interval. The simplest possible choice is the rectangular window.

Definition 1.1 (Rectangular window). *The rectangular window $\vec{\pi} \in \mathbb{R}^N$ with width w is defined as*

$$\vec{\pi}[j] := \begin{cases} 1 & \text{if } |j| \leq w, \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

The rectangular window does not distort the signal at all within the segment over which it is nonzero. However, we are interested in obtaining a local frequency representation, so we need to evaluate whether it also preserves the structure of the signal in the frequency domain. To study this we consider a discrete signal that consists of a single sinusoid. Ideally, when we multiply such a signal by a window function, we would like its DFT to remain as close to the DFT of the sinusoid as possible. The following theorem shows that performing pointwise multiplication of a signal with a complex sinusoid in the time domain is equivalent to applying a shift to its DFT coefficients that depends on the frequency of the sinusoid.

Theorem 1.2 (Multiplication with a complex sinusoid). *Let $x \in \mathbb{C}^N$ be an arbitrary N -dimensional*

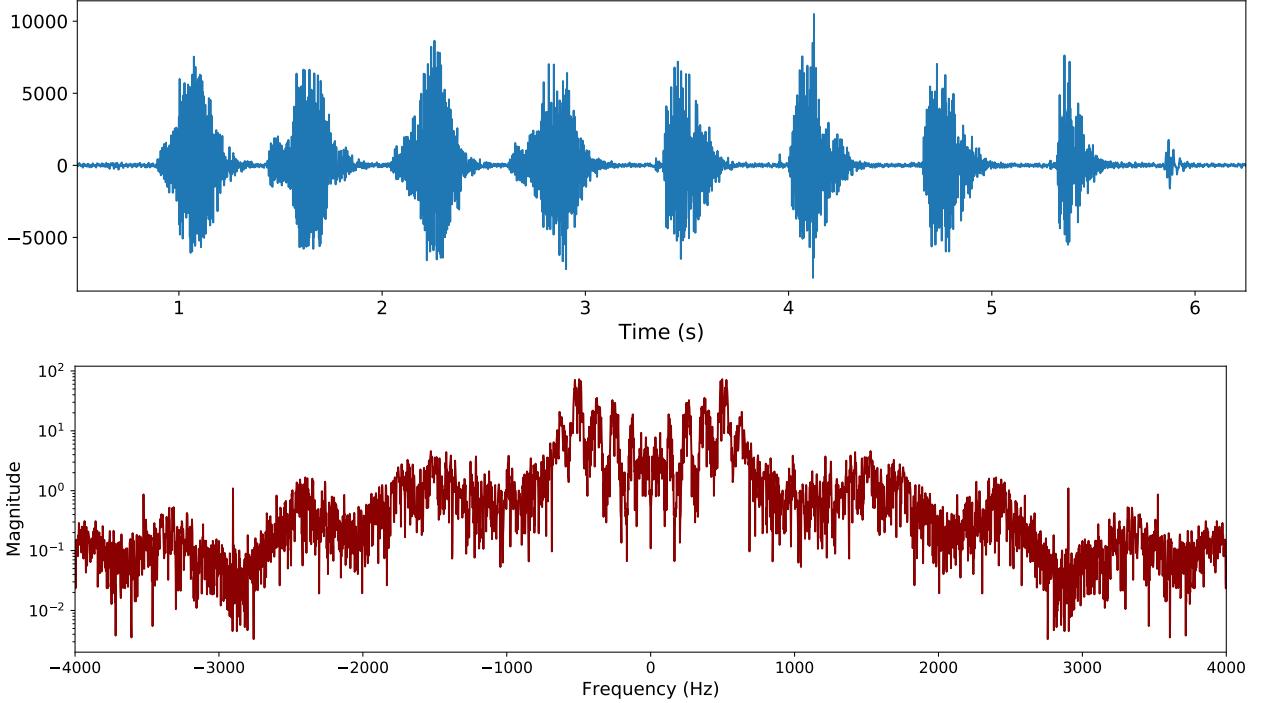


Figure 1: The top plot shows an audio signal of a person saying the word *no, no, no, no, yes, yes, yes, yes* in Hebrew. The bottom plot shows the corresponding Fourier series coefficients.

discrete signal, and $\psi_{k^*} \in \mathbb{C}^N$ the discrete complex N -dimensional sinusoid with frequency k^* ,

$$\psi_{k^*}[j] := \exp\left(\frac{i2\pi k^* j}{N}\right) \quad \text{for } 0 \leq j \leq N-1. \quad (2)$$

The DFT coefficients of their pointwise multiplication $y := x \circ \psi_{k^*}$ equal

$$\hat{y}[k] = \hat{x}[k - k^*], \quad 0 \leq k \leq N, \quad (3)$$

where \hat{x} is the DFT of x .

Proof. We have

$$\hat{y}[k] := \sum_{j=1}^N x[j] \psi_{k^*}[j] \exp\left(-\frac{i2\pi k j}{N}\right) \quad (4)$$

$$= \sum_{j=1}^N x[j] \exp\left(\frac{i2\pi k^* j}{N}\right) \exp\left(-\frac{i2\pi k j}{N}\right) \quad (5)$$

$$= \sum_{j=1}^N x[j] \exp\left(\frac{-i2\pi (k - k^*) j}{N}\right) \quad (6)$$

$$= \hat{x}[k - k^*]. \quad (7)$$

□

An important consequence of the theorem is that when we window a signal, its DFT suddenly becomes a sum of shifts of the DFT of the window. The sum of shifted copies of a signal is called a convolution; we will talk more about this operation later on. The following corollary illustrates the effect of windowing on a real sinusoid.

Corollary 1.3 (Multiplication with a real sinusoid). *Consider the discrete real sinusoid*

$$s[j] := \cos\left(\frac{2\pi k^* j}{N}\right), \quad 0 \leq j \leq N. \quad (8)$$

The DFT coefficients of the pointwise multiplication between s and a signal $x \in \mathbb{C}^N$, $y := x \circ s$ equal

$$\hat{y}[k] = \frac{\hat{x}[k - k^*] + \hat{x}[k + k^*]}{2}, \quad 0 \leq k \leq N, \quad (9)$$

where \hat{x} is the DFT of x .

Proof. We have $s = \frac{\psi_{k^*} + \psi_{-k^*}}{2}$. The DFT is linear (it can be expressed as multiplication with a matrix), so the DFT of

$$xs = \frac{x\psi_{k^*} + x\psi_{-k^*}}{2} \quad (10)$$

is equal to the average of the DFTs of $x\psi_{k^*}$ and $x\psi_{-k^*}$. The result then follows from the theorem. \square

To understand how applying a rectangular window affects the frequency representation of a signal, we need to derive its DFT. To simplify the exposition we index the vector and its DFT from $-N/2 + 1$ to $N/2$, assuming that N is even.

Lemma 1.4. *The DFT coefficients of the rectangular window $\vec{\pi} \in \mathbb{C}^N$ with width $2w$ from Definition 1.1*

$$\vec{\pi}[j] := \begin{cases} 1 & \text{if } |j| \leq w, \\ 0 & \text{otherwise,} \end{cases} \quad (11)$$

equals the discrete sinc or Dirichlet kernel

$$\hat{\pi}[k] = \begin{cases} 2w + 1 & \text{for } k = 0 \\ \frac{\sin\left(\frac{2\pi k(w+1/2)}{N}\right)}{\sin\left(\frac{\pi k}{N}\right)}, & \text{for } k \in \{-N/2 + 1, -N/2 + 2, \dots, N/2\} / 0, \end{cases} \quad (12)$$

Proof. We have

$$\hat{\pi}[0] = \sum_{j=-N/2+1}^{N/2} \vec{\pi}[j] \quad (13)$$

$$= \sum_{j=-w}^w 1 \quad (14)$$

$$= 2w + 1. \quad (15)$$

For $k \neq 0$,

$$\hat{\pi}[k] = \sum_{j=-N/2+1}^{N/2} \vec{\pi}[j] \exp\left(-\frac{i2\pi kj}{N}\right) \quad (16)$$

$$= \sum_{j=-w}^w \exp\left(-\frac{i2\pi k j}{N}\right)^j \quad (17)$$

$$= \frac{\exp\left(\frac{i2\pi kw}{N}\right) - \exp\left(-\frac{i2\pi k(w+1)}{N}\right)}{1 - \exp\left(-\frac{i2\pi k}{N}\right)} \quad (18)$$

$$= \frac{\exp\left(-\frac{i2\pi k}{2N}\right) 2i \sin\left(\frac{2\pi k(w+1/2)}{N}\right)}{\exp\left(-\frac{i2\pi k}{2N}\right) 2i \sin\left(\frac{\pi k}{N}\right)} \quad (19)$$

$$= \frac{\sin\left(\frac{2\pi k(w+1/2)}{N}\right)}{\sin\left(\frac{\pi k}{N}\right)}. \quad (20)$$

□

Unfortunately, the discrete sinc has side lobes and does not decay rapidly (see Figure 2). As a result, windowing a sinusoid produces significant distortion in the frequency domain, as shown in Figure 2. The DFT of the full sinusoid consists of two spikes at the corresponding frequency and its negative. In contrast, the DFT of the windowed sinusoid consists of two sinc functions centered at those frequencies. Due to the side lobes and slow decay, this makes determining the original frequency challenging, especially if there are other components of noise.

The distortion introduced by windowing with a rectangular window is exacerbated by the discontinuity in the rectangular window. It introduces an artificial discontinuity in the windowed function, which gives rise to spurious high-frequency components in its DFT. In order to alleviate this, we can instead use a tapered window, which decreases smoothly to zero at the borders. The Hann window is a popular choice. As we can see in Figure 2, this window produces much less distortion in the frequency domain, because its DFT decays faster as a result of its smoothness. Note however, that it distorts the original signal more in the time domain. There is therefore a tradeoff between distortion in time and frequency when windowing a signal.

Definition 1.5. *The Hann window $h \in \mathbb{R}^N$ of width $2w$ equals*

$$h[j] := \begin{cases} \frac{1}{2} \left(1 + \cos\left(\frac{\pi j}{w}\right)\right) & \text{if } |j| \leq w, \\ 0 & \text{otherwise.} \end{cases} \quad (21)$$

Multiplying by a Hann window (or other similar options) and then computing the DFT is a popular technique for analyzing the frequency content of signals locally. The time resolution of the analysis is governed by the width of the window. Reducing the window increases the resolution in time, but there is a price to pay. The following theorem shows that compressing a signal in time, dilates it in frequency, and vice versa.

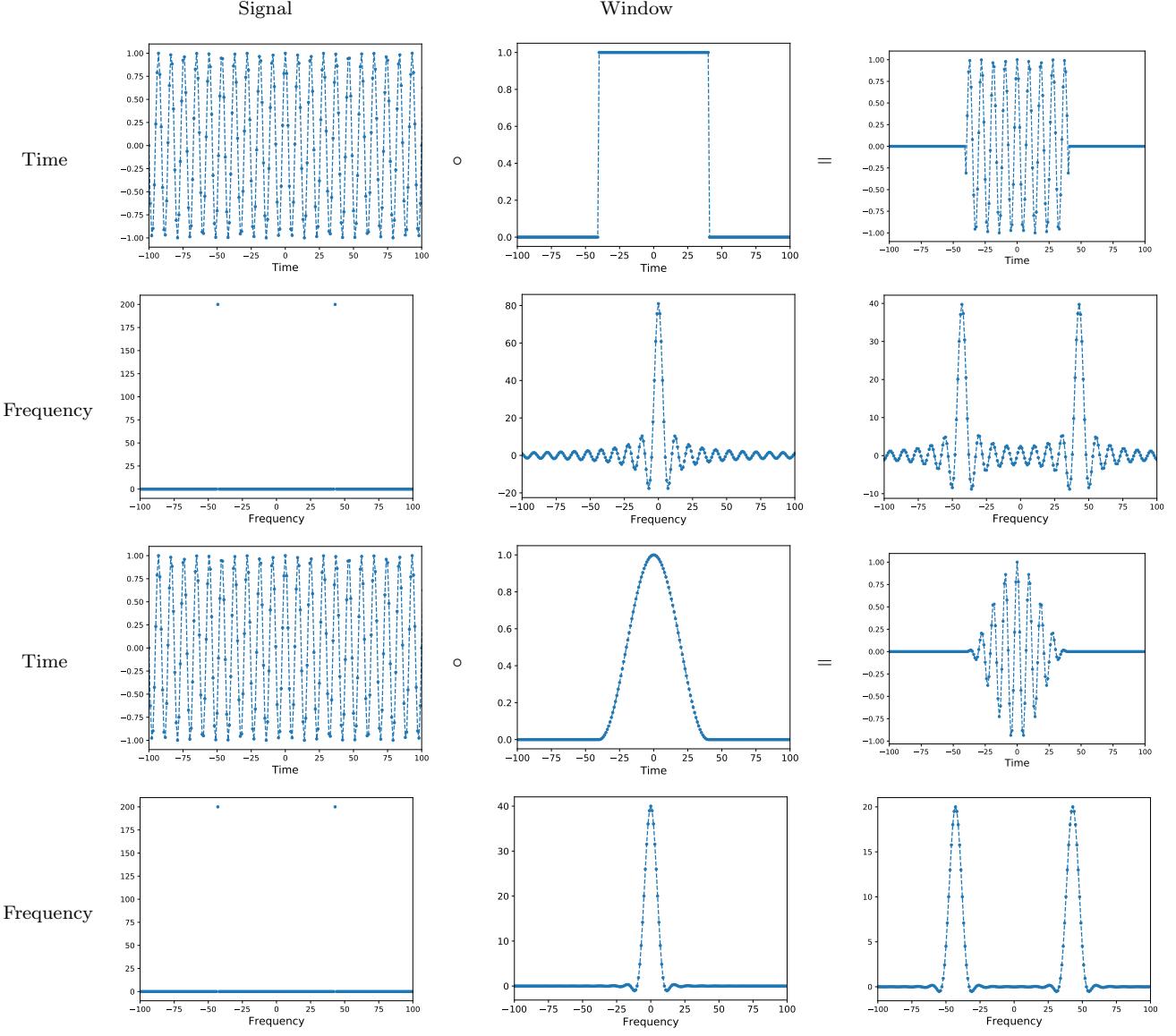


Figure 2: The left column shows a discrete real sinusoid and its DFT coefficients. The middle column shows a rectangular (top two rows) and a Hann window (bottom two rows) and their corresponding DFTs. When the sinusoid is multiplied with the windows (right column), the DFT of the resulting signal corresponds to two shifted copies of the DFT of the window as established in Corollary 1.3.

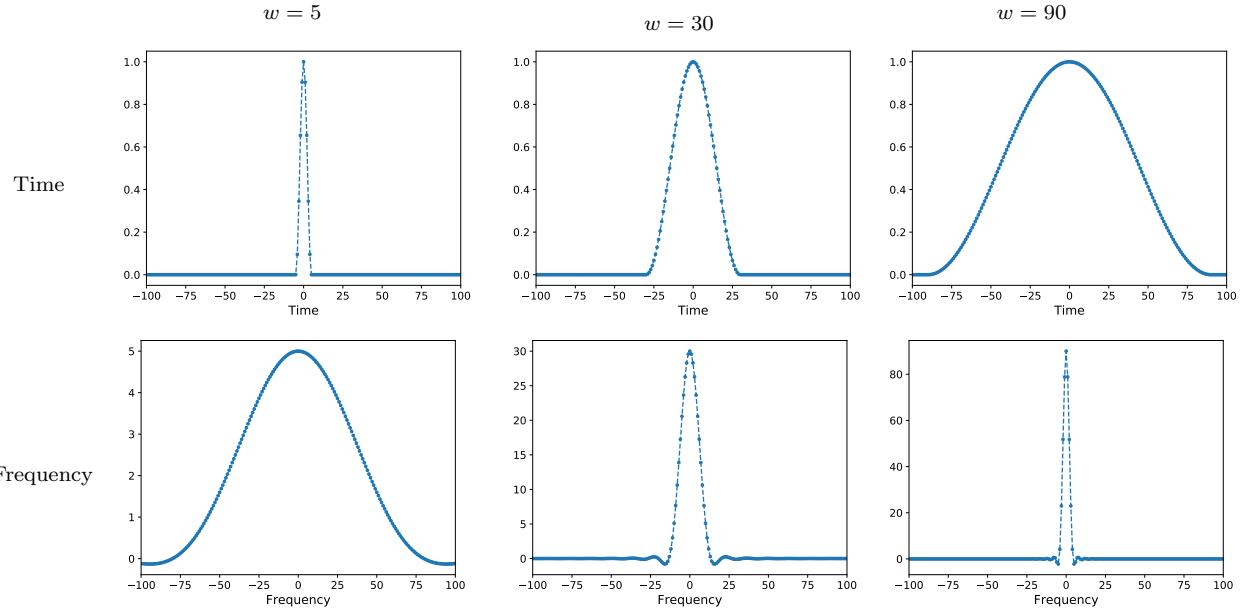


Figure 3: Dilations and contractions of a Hann window in time and the corresponding effect on the DFT of the window.

Theorem 1.6. Let $x \in \mathcal{L}_2[-T/2, T/2]$, $T > 0$, which is nonzero only in a band of width $2w$ around zero, i.e. $x(t) = 0$ for $t > |w|$ where $w > 0$. The Fourier series coefficients of the signal y that satisfies

$$y(t) = x(\alpha t), \quad \text{for all } t \in [-T/2, T/2], \quad (22)$$

equal

$$\hat{y}[k] = \frac{1}{\alpha} \langle x, \phi_{k/\alpha} \rangle \quad (23)$$

for any positive real number α such that $w/\alpha < T/2$.

Proof. By definition of the Fourier series coefficients

$$\hat{y}[k] = \int_{t=-T/2}^{T/2} y(t) \exp\left(-\frac{i2\pi kt}{T}\right) dt \quad (24)$$

$$= \int_{t=-w/\alpha}^{w/\alpha} x(\alpha t) \exp\left(-\frac{i2\pi kt}{T}\right) dt \quad (25)$$

$$= \frac{1}{\alpha} \int_{\tau=-w}^w x(\tau) \exp\left(-\frac{i2\pi k\tau}{\alpha T}\right) d\tau \quad (26)$$

$$= \frac{1}{\alpha} \int_{\tau=-T/2}^{T/2} x(\tau) \exp\left(-\frac{i2\pi k\tau}{\alpha T}\right) d\tau. \quad (27)$$

□

There is consequently a fundamental trade-off between the resolution in time and frequency that we can achieve simultaneously. If we make a window narrower to increase the time resolution, its DFT will become wider. As a result, in the frequency domain the multiplication with the window will blur the frequency content of the signal to a greater extent, decreasing the frequency resolution of the analysis. This fundamental tradeoff is known as the *uncertainty principle*. We refer the interested reader to Section 2.3.2 of Ref. [1] for more details.

1.2 Short-time Fourier transform

As discussed in the previous section, 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. The short-time Fourier transform (STFT) is designed to describe localized fluctuations. The STFT coefficients are equal to the DFT of time segments of the signal, extracted through multiplication with a window. Here we focus on the discrete STFT; it is also possible to define continuous versions.

Definition 1.7 (Short-time Fourier transform). *The short-time Fourier transform of a vector $x \in \mathbb{C}^N$ is given by*

$$\text{STFT}_{[\ell]}(x)[k, s] := \left\langle x, \xi_k^{\downarrow s(1-\alpha_{\text{ov}})\ell} \right\rangle, \quad 0 \leq k \leq \ell - 1, \quad 0 \leq s \leq \frac{N-1}{(1-\alpha_{\text{ov}})\ell}, \quad (28)$$

where the basis vectors are defined as

$$\xi_k[j] := \begin{cases} w_{[\ell]}(j) \exp\left(\frac{i2\pi kj}{\ell}\right) & \text{if } 1 \leq j \leq \ell \\ 0 & \text{otherwise,} \end{cases} \quad (29)$$

and $\xi_k^{\downarrow m}$ denotes the vector ξ_k shifted by m . The overlap between adjacent segments equals $\alpha_{\text{ov}}\ell$. The STFT is parametrized by the choice of segment length ℓ , window function $w_{[\ell]} \in \mathbb{C}^\ell$, and overlap factor α_{ov} .

The STFT coefficients are defined as the inner product between the signal and basis vectors corresponding to windowed shifted sinusoids. Figure 4 shows several basis vectors for a specific example.

Due to the overlap between the shifted signals in time, the STFT is an *overcomplete* transformation. If the overlap factor is $\alpha_{\text{ov}} := 0.5$ then the dimensionality of the STFT is approximately double (up to border effects) that of the original vector. This is apparent in the following matrix representation of the STFT,

$$\begin{bmatrix} \text{STFT}_{[\ell]}(x)[0, 0] \\ \dots \\ \text{STFT}_{[\ell]}(x)[\ell-1, 0] \\ \text{STFT}_{[\ell]}(x)[0, 1] \\ \dots \\ \text{STFT}_{[\ell]}(x)[\ell-1, 1] \\ \text{STFT}_{[\ell]}(x)[0, 2] \\ \dots \\ \text{STFT}_{[\ell]}(x)[\ell-1, 2] \\ \dots \end{bmatrix} := \begin{bmatrix} F_{[\ell]} & 0 & 0 & 0 & \dots \\ 0 & F_{[\ell]} & 0 & 0 & \dots \\ 0 & 0 & F_{[\ell]} & 0 & \dots \\ 0 & 0 & 0 & F_{[\ell]} & \dots \\ 0 & 0 & 0 & 0 & \dots \\ \dots & \dots & \dots & \dots & \dots \end{bmatrix} \begin{bmatrix} \text{diag}(w_{[\ell]}) & 0 & 0 & \dots \\ 0 & \text{diag}(w_{[\ell]}) & 0 & \dots \\ 0 & 0 & \text{diag}(w_{[\ell]}) & \dots \\ 0 & 0 & 0 & \text{diag}(w_{[\ell]}) & \dots \\ \dots & \dots & \dots & \dots & \dots \end{bmatrix} x,$$

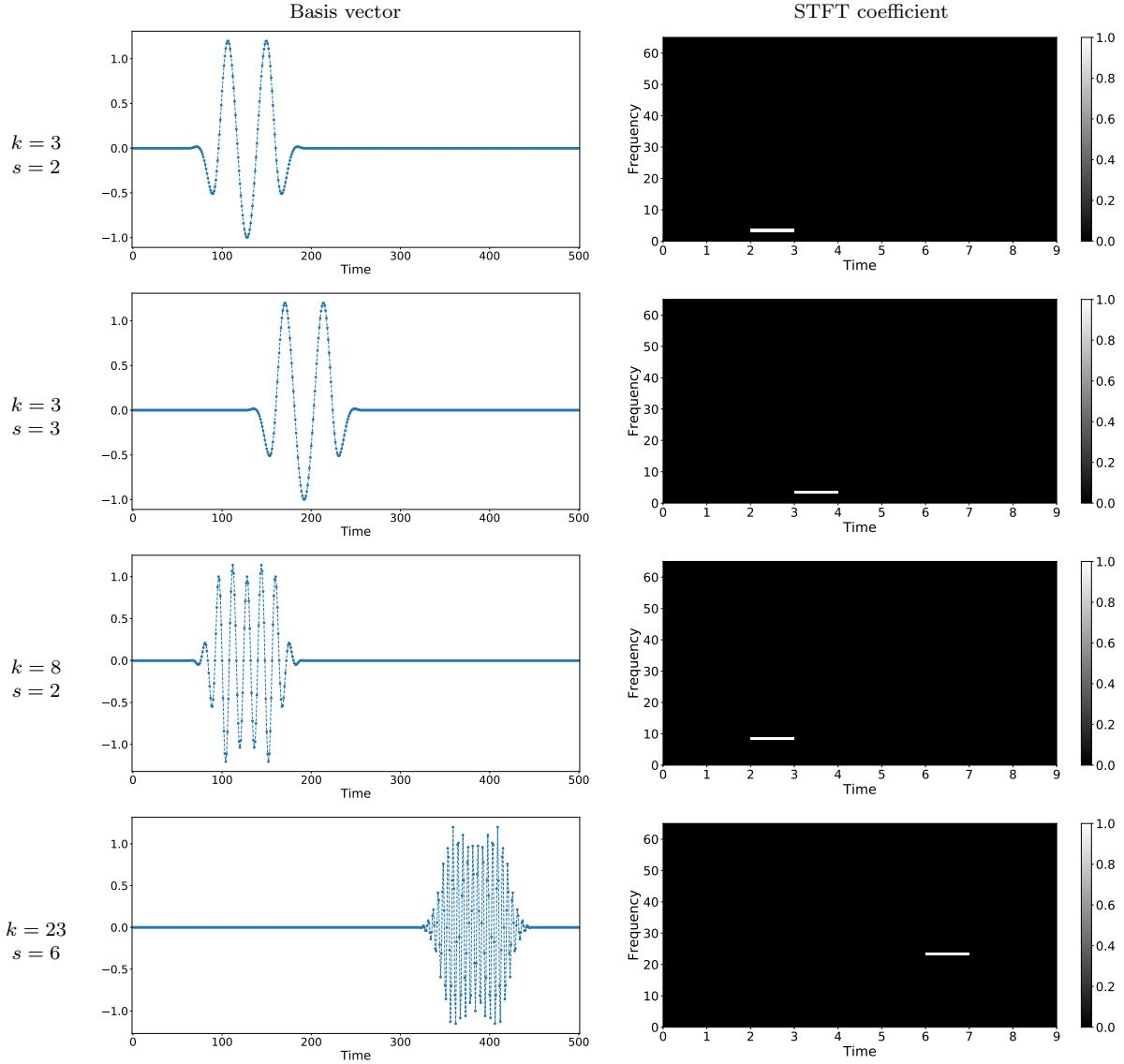


Figure 4: The left columns shows the real part of different basis vectors of the STFT for a signal of length $N := 500$ when $\ell := 128$ and $\alpha_{\text{ov}} := 0.5$ and the window is a Hann window. The right column shows the position of the corresponding STFT coefficient in a time-frequency grid. Only the positive frequency axis is shown.

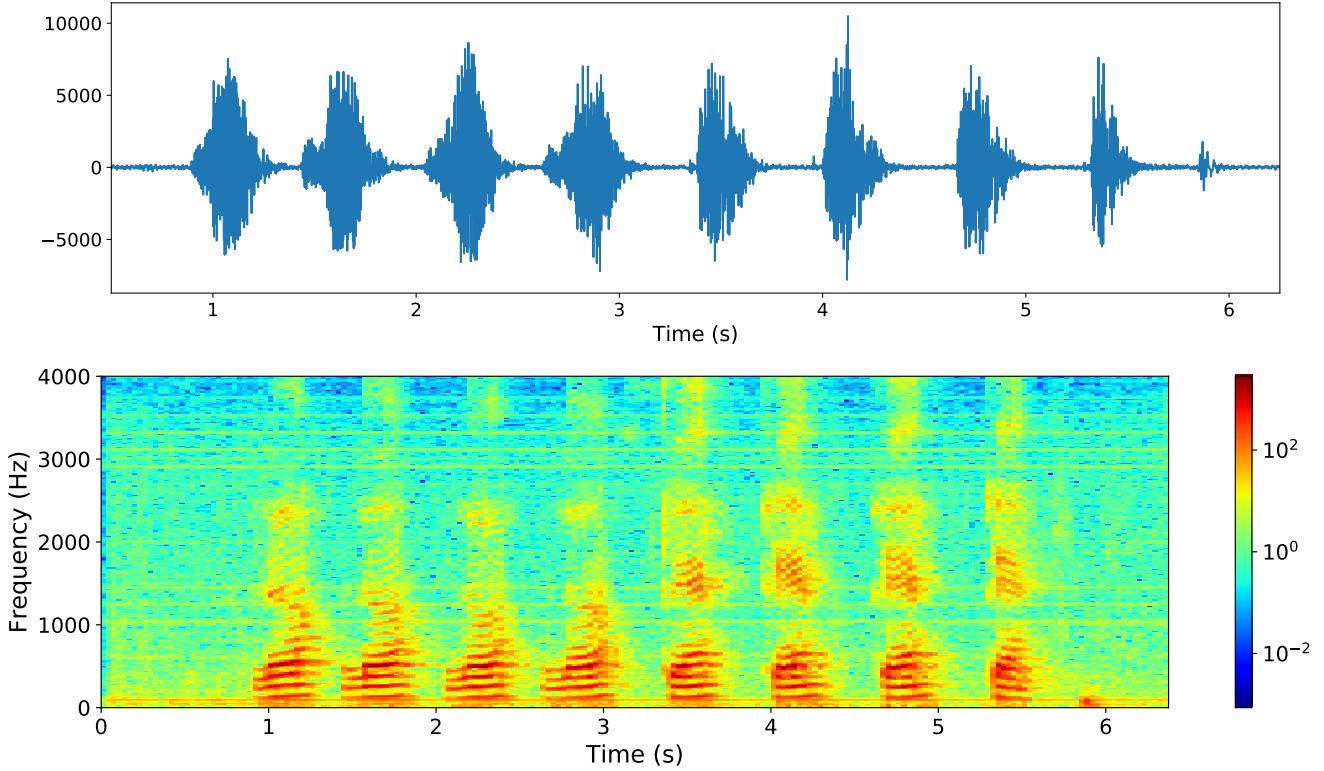


Figure 5: The figure shows the same signal as in Figure 1, along with a heatmap of the magnitude of its STFT coefficient computed using Hann windows of length 62.5 ms. Only the positive frequency axis is shown. In contrast to the Fourier series, the STFT visualizes when different frequency components are active.

where $F_{[\ell]}$ is an $\ell \times \ell$ DFT matrix, 0 represents a $\ell/2 \times \ell/2$ matrix full of zeros, and $\text{diag}(w_{[\ell]})$ is a diagonal matrix that has the window function as its diagonal. Overcomplete representations are known as *frames* in signal processing.

The STFT can be computed very efficiently using the FFT algorithm to obtain the DFT of the signal segments. The complexity of multiplying a segment by a window and applying the FFT is $O(\ell \log \ell)$. The number of segments is approximately $N/(1 - \alpha_{\text{ov}})\ell$ (ignoring the borders). The complexity of computing the STFT of a vector of length N is therefore $O(N \log \ell)$ (the overlap factor is usually equal to 0.5 or a similar fraction). To recover a signal from its STFT coefficients we can compute the inverse DFT of each segment, again applying the FFT algorithm, and then combine the scaled segments. The complexity is also $O(N \log \ell)$.

Figure 5 shows the magnitude of the STFT coefficients of a real-world speech signal. This representation, known as the *spectrogram*, is a fundamental tool in audio analysis. It is used to visualize how the signal energy in each frequency band evolves over time.

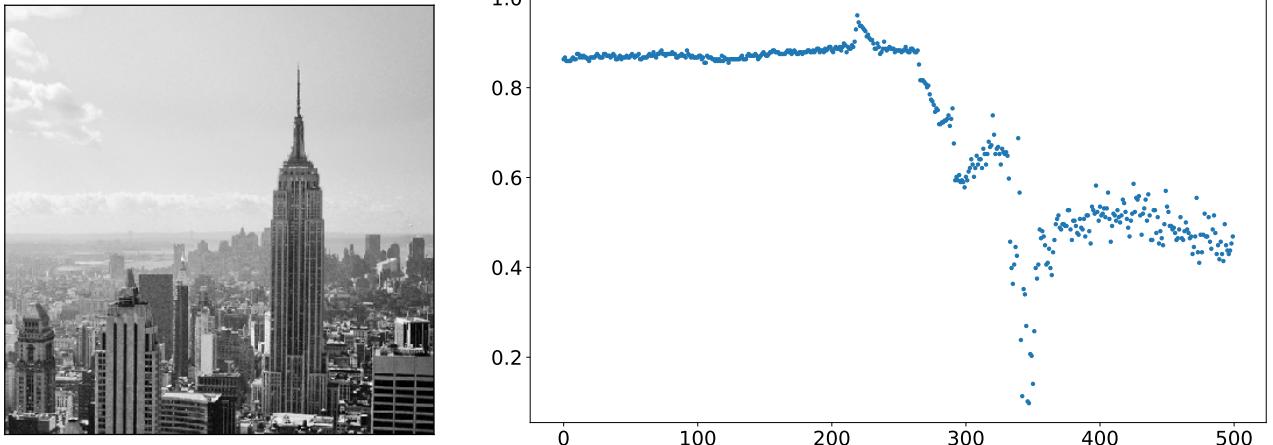


Figure 6: An image of New York City and the values of a subset of pixels situated on a vertical line (if the image is interpreted as a 500×500 image the pixels correspond to column 135).

2 Multiresolution analysis

2.1 Wavelets

The scale at which information is encoded in signals is usually not uniform. For example, the top half of the image in Figure 6 is very smooth, whereas the bottom half has a lot of high-resolution details. The aim of multiresolution analysis is to reveal this structure, by decomposing signals into components corresponding to different scales. To make this precise, we define a decomposition of the ambient signal space into subspaces, each of which encodes information at a certain resolution. This is a discrete version of the framework introduced by Mallat and Meyer (see Chapter 7 in [1]).

Definition 2.1 (Multiresolution decomposition). *Let $N := 2^K$ for some integer K . A multiresolution decomposition of \mathbb{R}^N is a sequence of nested subspaces $\mathcal{V}_K \subset \mathcal{V}_{K-1} \subset \dots \subset \mathcal{V}_0$. The subspaces must satisfy the following properties:*

- $\mathcal{V}_0 = \mathbb{R}^N$, the approximation at scale $2^0 = 1$ is perfect.
- \mathcal{V}_k is invariant to translations of scale 2^{k+1} for $0 \leq k \leq K$. If $x \in \mathcal{V}_k$ then

$$x^{\downarrow l2^{k+1}} \in \mathcal{V}_k \quad \text{for all } l \in \mathbb{Z}, \quad (30)$$

where $x^{\downarrow m}$ denotes the vector x translated by m .

- Dilating vectors in \mathcal{V}_j by 2 yields vectors in \mathcal{V}_{j+1} . Let $x \in \mathcal{V}_j$ be nonzero only between 1 and $N/2$, the dilated vector $x_{\leftrightarrow 2}$ belongs to \mathcal{V}_{j+1} (see Def. 2.2 below).

Definition 2.2 (Discrete dilation). *Let vector $x \in \mathbb{R}^N$, with entries indexed between 0 and $N - 1$, satisfy $x[j] = 0$ for all $j \geq N/M$, where M is a fixed positive integer. We define the dilation of x by a factor of M as*

$$x_{\leftrightarrow M}[j] = x \left[\left\lceil \frac{j}{M} \right\rceil \right]. \quad (31)$$

Intuitively, subspace \mathcal{V}_j contains information encoded at a scale 2^j . If we dilate a signal in \mathcal{V}_j by a factor of 2, thereby doubling its scale and halving its resolution, then it is guaranteed to belong to \mathcal{V}_{j+1} . By projecting a signal onto \mathcal{V}_j we obtain an approximation at the corresponding resolution. An example of a family of nested subspaces that satisfies these conditions are the vectors that are piecewise constant on segments of length 2^k for different values of k . If we translate a such a vector, then it is still piecewise constant. If we dilate such a vector it will be constant on segments of length 2^{k+1} .

Mallat and Meyer suggested the following strategy to build multiresolution decompositions:

- Set the coarsest subspace to be spanned by a low-frequency vector φ , called a scaling vector or father wavelet:

$$\mathcal{V}_1 := \text{span}(\varphi). \quad (32)$$

- Decompose the finer subspaces into the direct sum

$$\mathcal{V}_k := \mathcal{W}_{k+1} \oplus \mathcal{V}_{k+1}, \quad 0 \leq k \leq K-1, \quad (33)$$

where \mathcal{W}_{k+1} captures the finest resolution available at level k . Set \mathcal{W}_k to be spanned by shifts of a vector μ , called a mother wavelet, dilated to have the appropriate resolution:

$$\mathcal{W}_k := \bigoplus_{m=0}^{\frac{N-1}{2^{k+1}}} \text{span}\left(\mu_{\leftrightarrow 2^k}^{\downarrow m 2^{k+1}}\right). \quad (34)$$

- This yields the decomposition

$$\mathbb{R}^N = \mathcal{V}_0 \quad (35)$$

$$= \mathcal{V}_1 \oplus \mathcal{W}_1 \quad (36)$$

$$= \mathcal{V}_2 \oplus \mathcal{W}_2 \oplus \mathcal{W}_1 \quad (37)$$

$$= \mathcal{V}_k \oplus \mathcal{W}_k \oplus \cdots \oplus \mathcal{W}_2 \oplus \mathcal{W}_1. \quad (38)$$

\mathcal{W}_k is spanned by scaled and shifted copies of the mother wavelet, which capture information at the corresponding resolution. \mathcal{V}_1 captures the information at the coarsest resolution, and is spanned by the father wavelet (alternatively, it could be spanned by shifted copies of the father wavelet). Designing valid multiresolution decompositions requires selecting a specific father and mother wavelet pair, such that the corresponding subspaces satisfy the requirements. The Haar wavelets yield a multiresolution decomposition for piecewise constant vectors.

Definition 2.3 (Haar wavelet basis). *The Haar father wavelet $\varphi \in \mathbb{R}^N$ is a constant vector, such that*

$$\varphi[j] := \frac{1}{\sqrt{N}}, \quad 1 \leq j \leq N. \quad (39)$$

The mother wavelet $\mu \in \mathbb{R}^N$ satisfies

$$\mu[j] := \begin{cases} -\frac{1}{\sqrt{2}}, & j = 1, \\ \frac{1}{\sqrt{2}}, & j = 2, \\ 0, & j > 2. \end{cases} \quad (40)$$

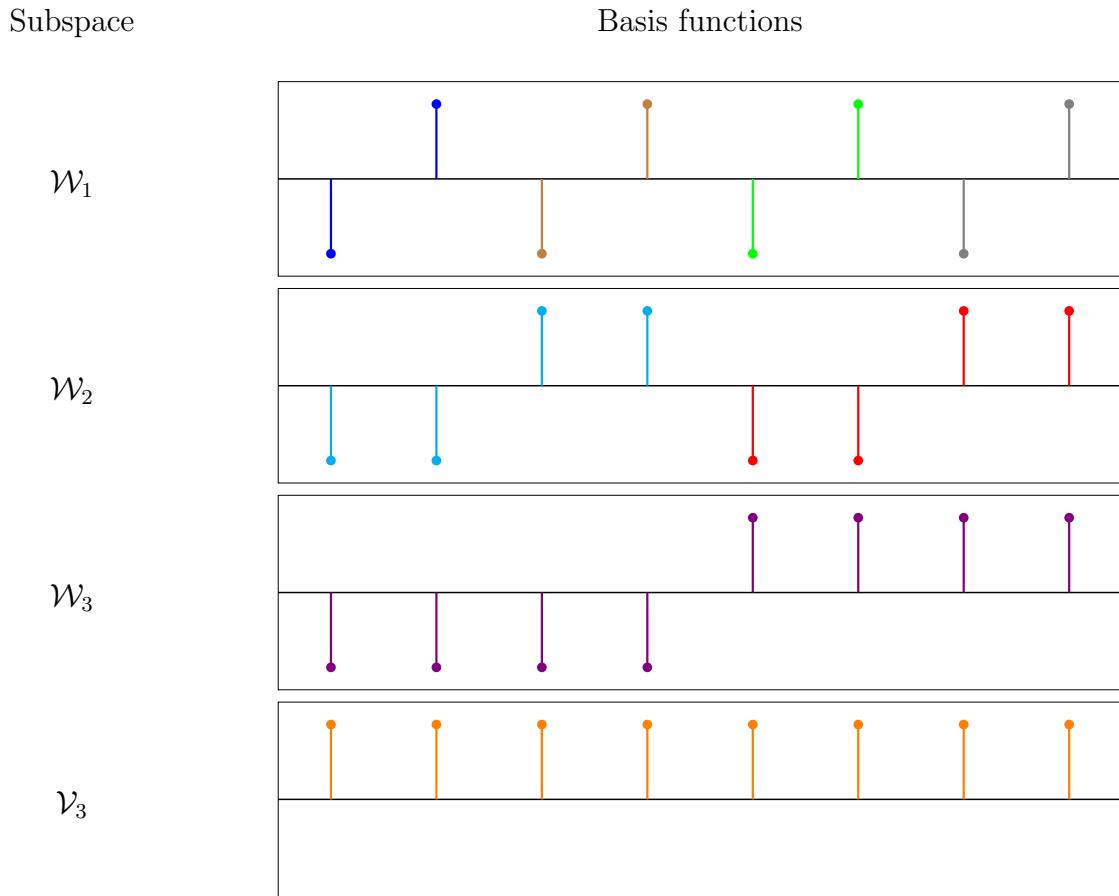


Figure 7: Basis vectors in the Haar wavelet basis for \mathbb{R}^8 . Each row shows the vectors corresponding to the corresponding scale in different colors.

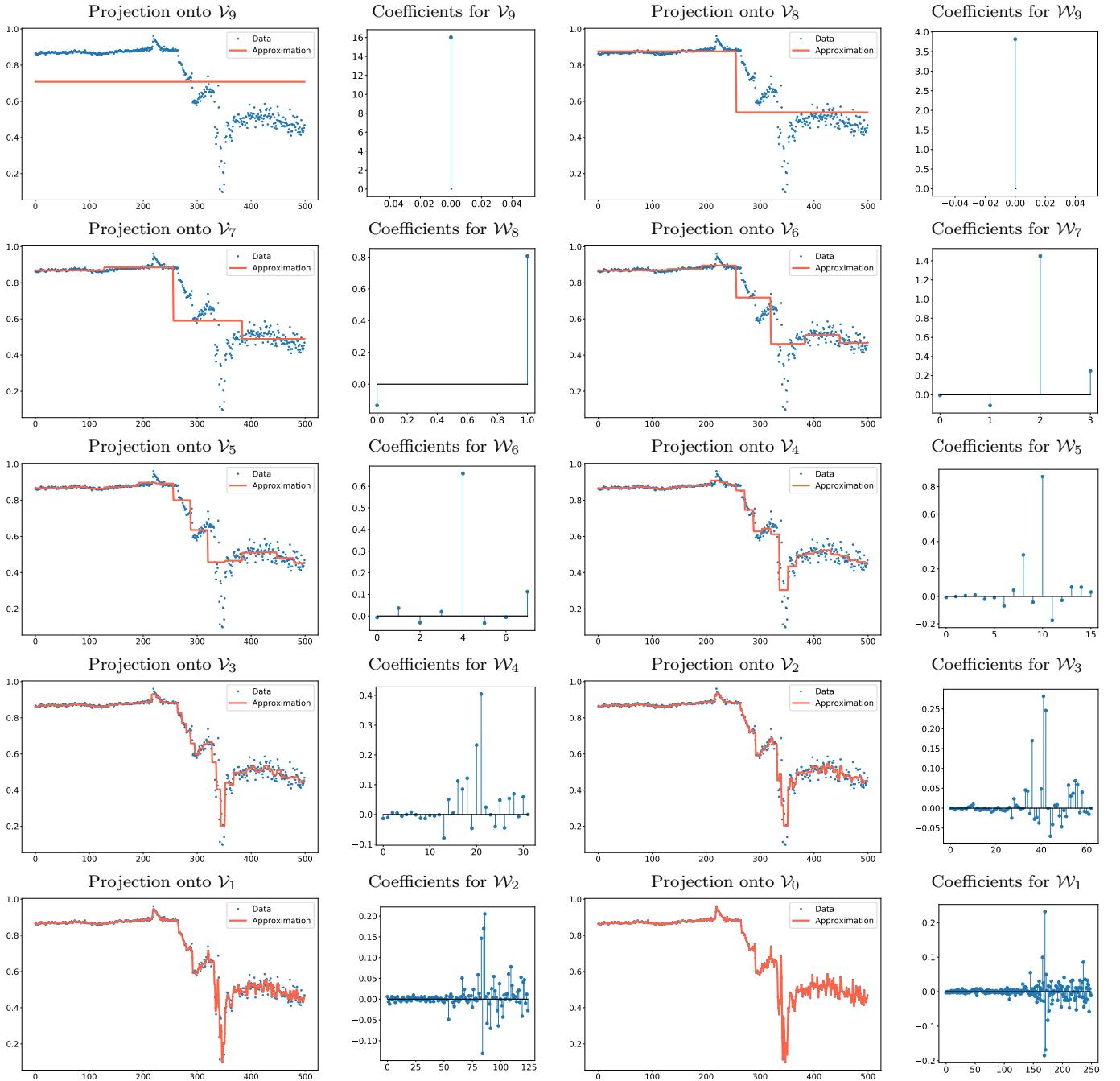


Figure 8: Multiresolution decomposition of the 1D signal corresponding to a vertical line of the image in Figure 6 using the 1D Haar wavelet basis. The additional wavelet coefficients incorporated at each scale are plotted on the right.

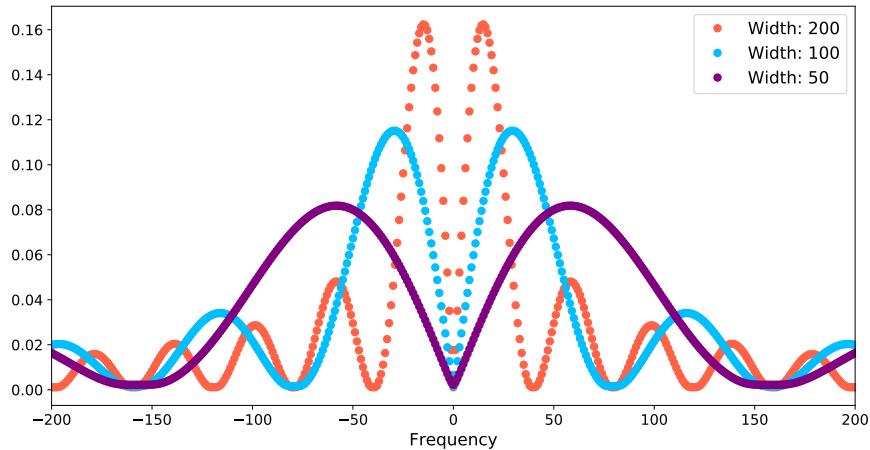


Figure 9: DFT of Haar wavelets with different widths.

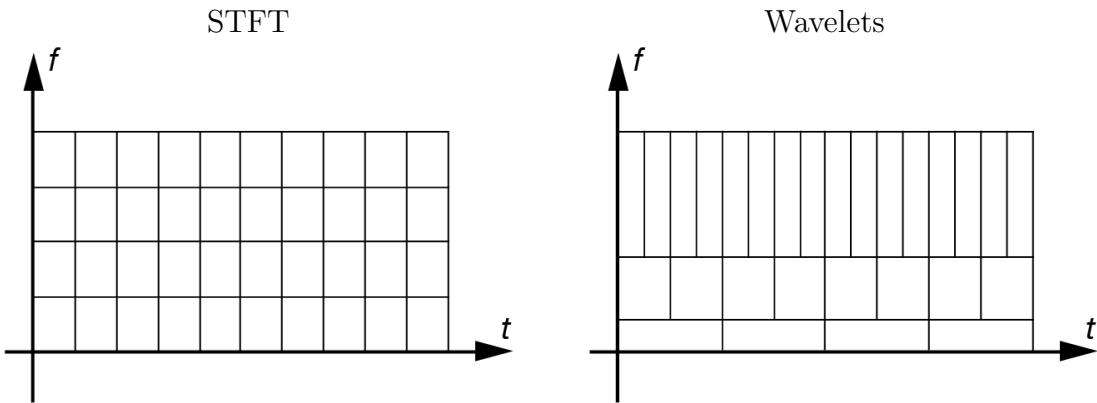


Figure 10: Diagram of the time-frequency support of STFT (left) and wavelet basis vectors (right).

One can check that the Haar wavelet provides a valid multiresolution decomposition (you will do this in the homework). \mathcal{V}_k corresponds to vectors that are piecewise constant on segments of length 2^k . The basis vectors are all orthogonal and unit norm when rescaled appropriately, so they can be used to form an orthonormal basis of \mathbb{R}^N . Figure 7 shows the basis vectors for $N = 8$. In Figure 8 the decomposition is applied to obtain a multiscale decomposition of the 1D signal corresponding to a vertical line of the image in Figure 6. Approximations obtained using the Haar wavelet are piecewise constant, which may not be desirable. It is also possible to build multiresolution decompositions with smoother wavelets. Examples include Meyer wavelets, Daubechies wavelets, coiflets, symmlets, etc. We refer the interested reader to Chapter 7 in [1] for a detailed description.

In contrast to the STFT, multiresolution analysis provides a time-frequency decomposition of signals where different coefficients have different resolutions. Figure 9 shows the DFT of Haar wavelets corresponding to different scales. As established in Theorem 1.6, when the wavelet contracts (decreasing the time resolution), its DFT dilates (increasing the frequency resolution). Note that the Haar wavelet is not very localized in frequency due to its discontinuity, just like the rectangular window (see Lemma 1.4). Figure 10 provides a cartoon illustration of the time-

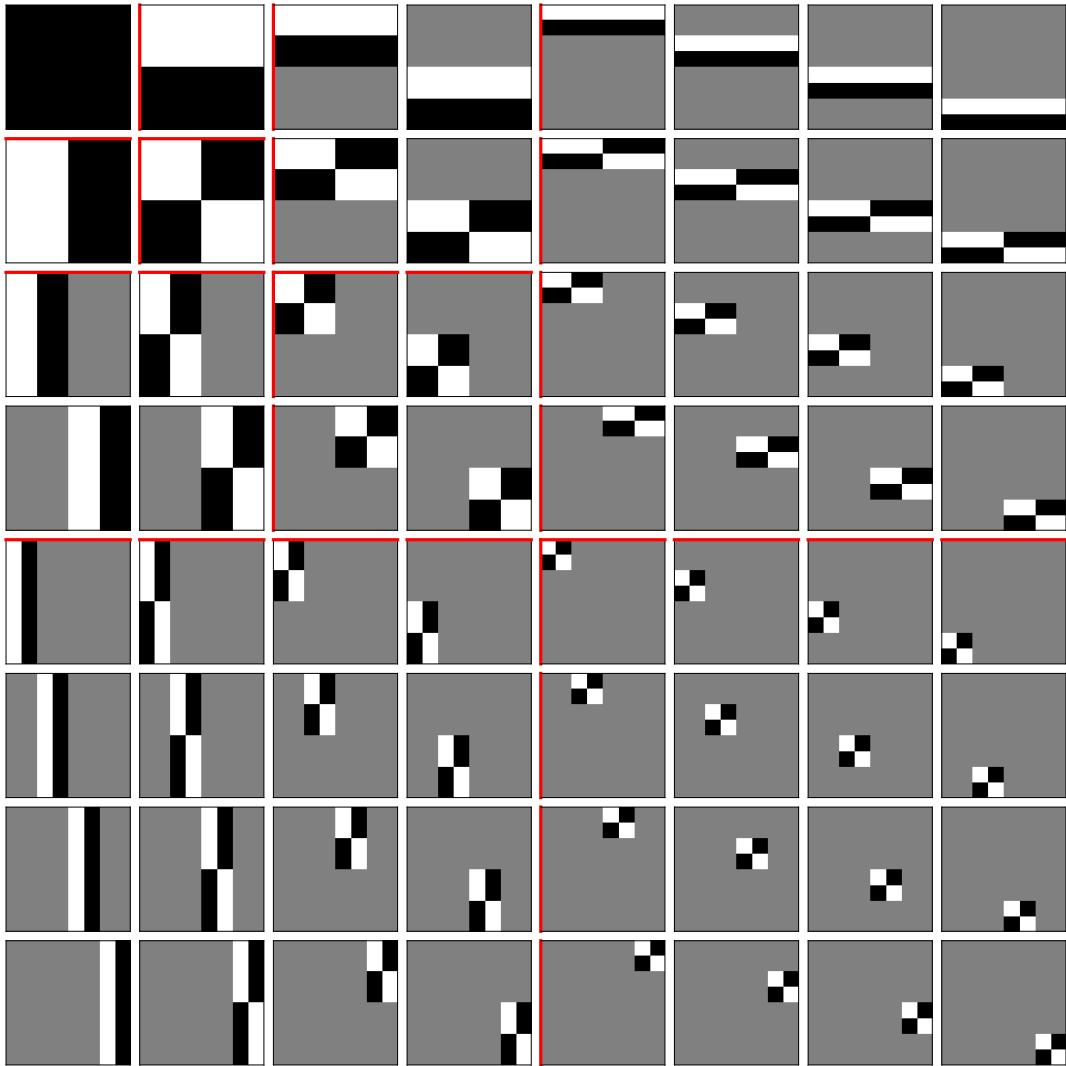


Figure 11: Basis vectors of the 2D Haar wavelet decomposition.

frequency supports of STFT and wavelet basis vectors. The STFT tiles the time-frequency plane regularly, whereas wavelet vectors have very different time and frequency resolutions.

2.2 Multidimensional wavelets

Multiresolution decompositions are an important tool in image processing, where they are applied to 2D and 3D signals. A simple way to design multidimensional decompositions is to leverage 1D multiresolution decompositions. The basis vectors of each subspace are constructed by taking outer products of the corresponding one-dimensional wavelets. Let $v_{[m,s]}^{1D}$ denote a basis vector at scale m shifted by s (depending on the resolution it can be a father or mother wavelet). To build a 2D basis vector at scale (m_1, m_2) and shift (s_1, s_2) we set

$$v_{[s_1, s_2, m_1, m_2]}^{2D} := v_{[s_1, m_1]}^{1D} (v_{[s_2, m_2]}^{1D})^T, \quad (41)$$

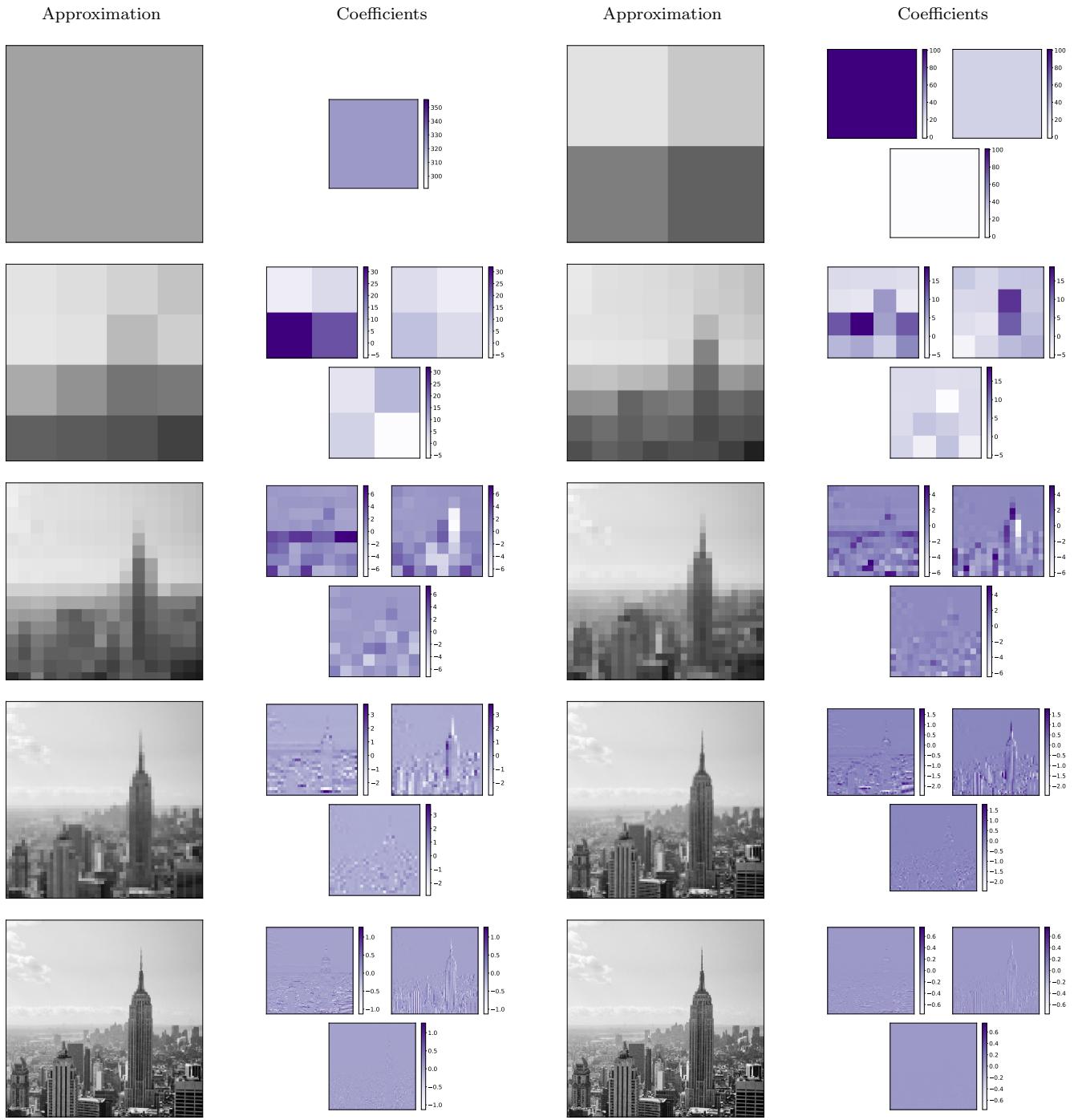


Figure 12: Multiresolution decomposition of the image in Figure 6 using the 2D Haar wavelet basis. The corresponding wavelet coefficients are plotted on the right for each scale.

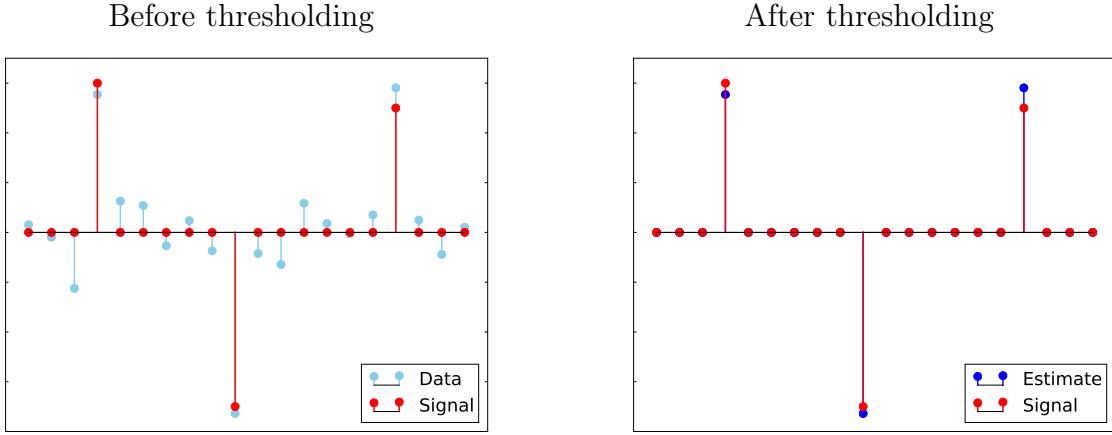


Figure 13: Denoising via hard thresholding.

where v^{1D} can refer to 1D father or mother wavelets.

Figure 11 shows the resulting 2D basis vectors for the Haar wavelet basis. Two-dimensional wavelet transforms are a popular tool to perform multiresolution decompositions of images. Figure 12 shows a multiresolution decomposition of the image in Figure 6 using the 2D Haar wavelet basis. One can use any 1D wavelets to build a 2D multiresolution decomposition. However, better performance can be achieved by designing non-separable basis vectors. Examples of non-separable 2D wavelets include the steerable pyramid, curvelets, and bandlets. See [here](#) and Section 9.3 in [1] for more details.

3 Denoising via thresholding

3.1 Hard thresholding

The STFT and wavelet transforms often yield sparse representations of signals, where many coefficients are close 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 5). 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 (see Figure 12).

In contrast, noisy perturbations usually have dense coefficients in any fixed frame or basis. A linear transformation $A\tilde{z}$ of a Gaussian vector with mean μ and covariance matrix Σ is still Gaussian with mean $A\mu$ and covariance matrix $A\Sigma A^*$. If A is an orthogonal matrix and the noise is iid Gaussian with mean zero and variance σ^2 , then $A\tilde{z}$ is also iid Gaussian with mean zero and variance σ^2 (the covariance matrix equals $A\sigma^2 I A^* = N\sigma^2 I$). This means, for example, that the Haar wavelet coefficients of the noise will not be sparse like the ones of a natural image.

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

additive noise $z \in \mathbb{C}^n$

$$y := x + z. \quad (42)$$

Under the assumptions that (1) Ax is sparse representation where A contains the basis vectors of a basis or a frame, and (2) the entries of Az are small and dense, eliminating small entries in the coefficients

$$Ay = Ax + Az \quad (43)$$

should therefore suppress the noise while preserving the signal. Figure 13 shows a simple example, where the signal itself is sparse (A is just the identity matrix).

Algorithm 3.1 (Denoising via hard thresholding). *Let y follow the model in equation (42) and let A be a full-rank linear transformation that sparsifies the signal x . To denoise we:*

1. *Compute the coefficients Ay .*
2. *Apply the hard-thresholding operator $\mathcal{H}_\eta : \mathbb{C}^n \rightarrow \mathbb{C}^n$ to the coefficients. Let $v \in \mathbb{C}^N$, the operator is defined as*

$$\mathcal{H}_\eta(v)[j] := \begin{cases} v[j] & \text{if } |v[j]| > \eta, \\ 0 & \text{otherwise,} \end{cases} \quad (44)$$

for $1 \leq j \leq N$. The threshold η can be adjusted according to the standard deviation of Az , or by cross validation.

3. *Compute the estimate by inverting the transform, i.e. setting*

$$x_{\text{est}} := L \mathcal{H}_\eta(Ay), \quad (45)$$

where L is a left inverse of A .

Note that thresholding is nonlinear: different inputs are scaled by different factors (1 or 0). This makes it possible to adapt to the local structure of the signal and usually results in more effective denoising. Figures 14, 15 and 16 show the results of denoising a speech signal applying hard thresholding. Figures 17 and 18 show the results of denoising an image by thresholding its 2D Haar wavelet coefficients. In both cases the threshold is chosen by cross validation on a separate set of signals.

3.2 Block thresholding

When we compute representations that capture localized details of signals, such as the wavelet transform or the STFT, the coefficients tend to be highly structured. Nonzero STFT coefficients of audio signals are often clustered near each other (see Figure 5). Similarly, nonzero wavelet coefficients of images are often concentrated close to edges and areas with high-frequency textures. This is apparent in Figure 12.

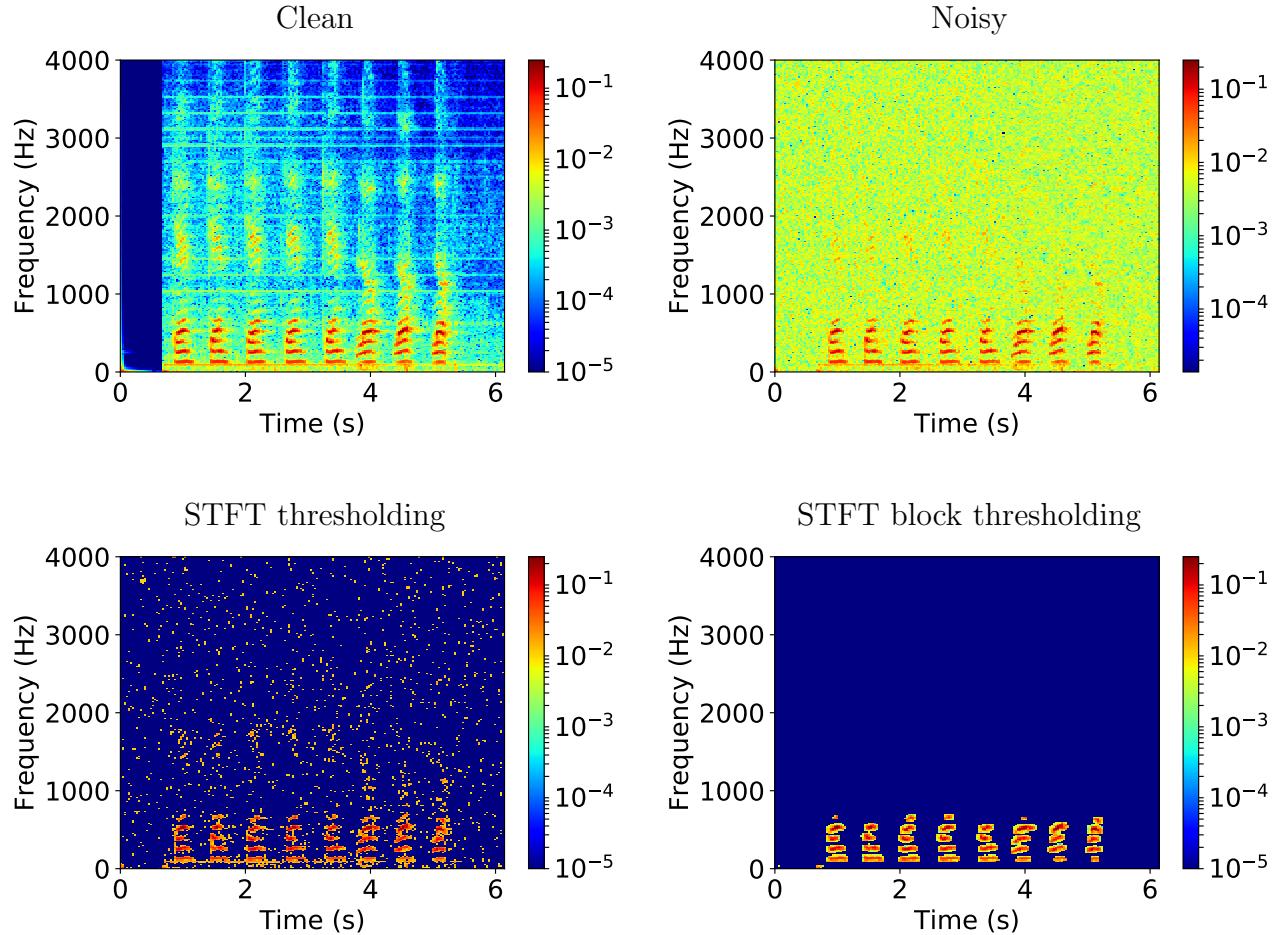


Figure 14: The top row shows the spectrogram of a test audio signal before and after being corrupted by additive iid Gaussian noise with standard deviation $\sigma = 0.1$. The bottom row shows the spectrogram of the signal denoised via STFT thresholding and block thresholding. Only the positive frequency axis is shown.

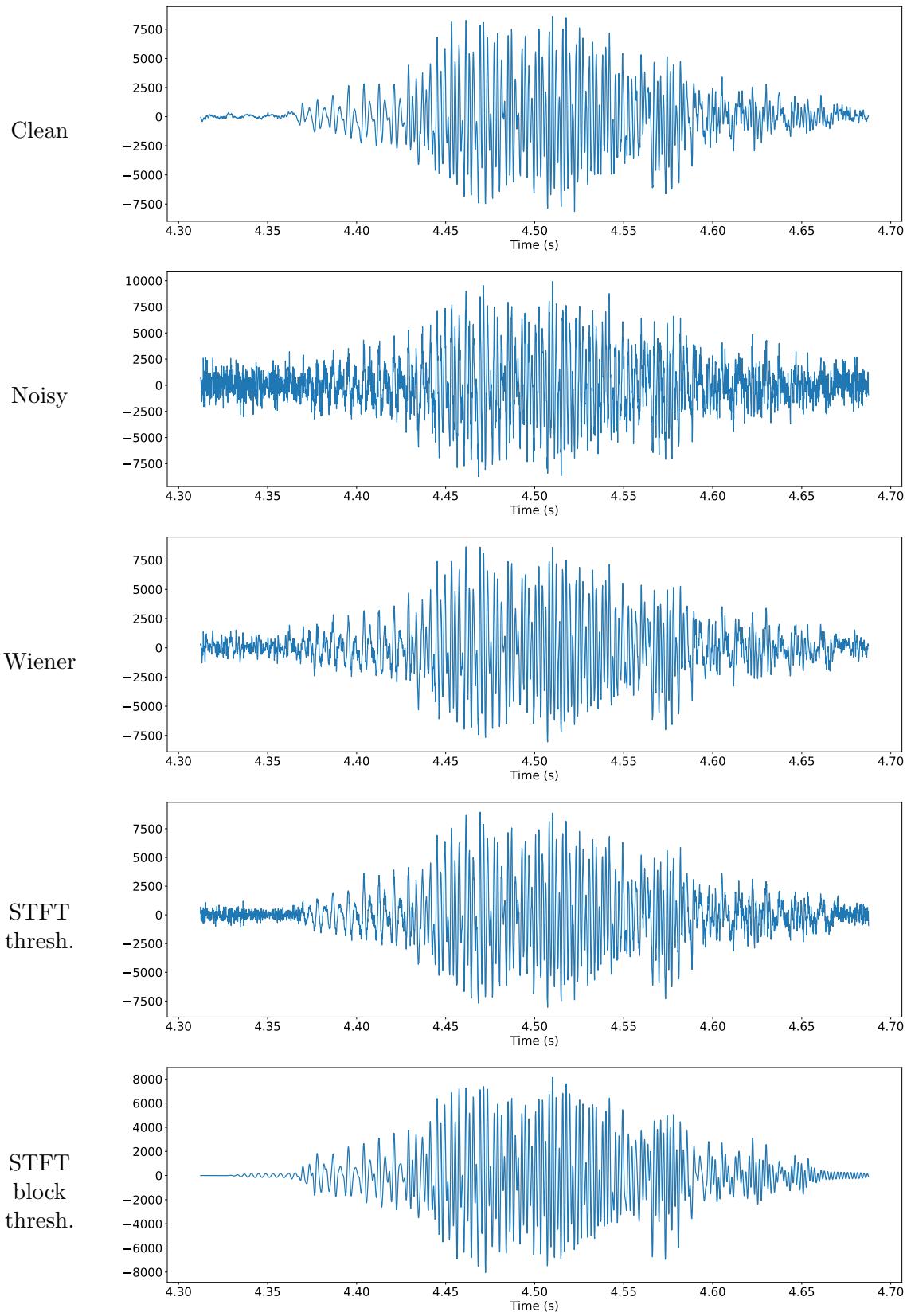


Figure 15: The top two images show the clean and noisy signals in Figure 14 in the time domain. Below, the signal denoised by Wiener filtering is compared to the result of applying STFT thresholding and block thresholding. Click on these links to listen to the audio: [clean signal](#), [noisy signal](#), denoised signal obtained by [Wiener filtering](#), [STFT thresholding](#), [STFT block thresholding](#).

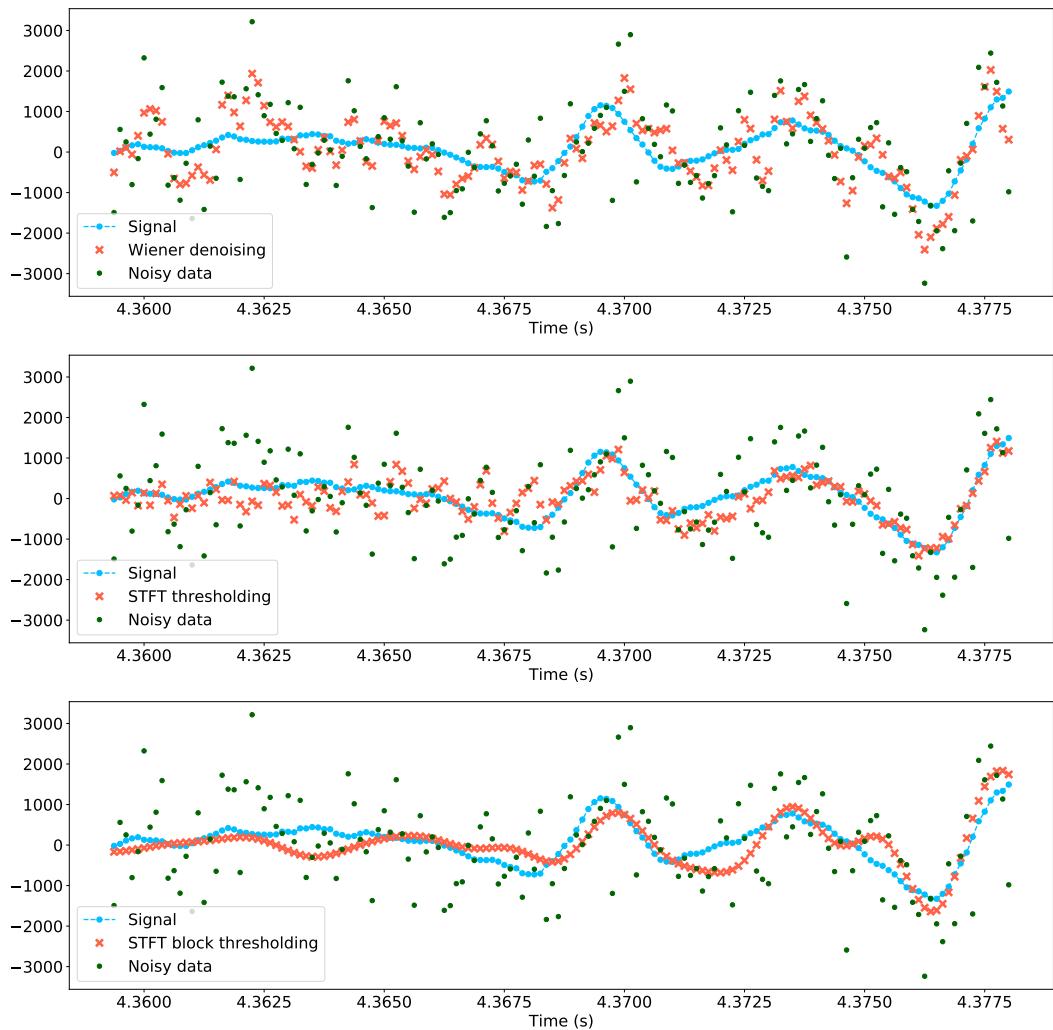


Figure 16: Zoomed-in plots of the signals in Figure 15.

Thresholding-based denoising can be enhanced by taking into account prior assumptions on coefficient structure. If we have a reason to believe that nonzero coefficients in the signal tend to be close to each other, then we should modify our thresholding scheme. Small coefficients that are isolated probably correspond to noise, and should be suppressed. However, small coefficients in the vicinity of large coefficients may contain useful information and should be left alone. This strategy is known as block thresholding.

Algorithm 3.2 (Denoising via block thresholding). *Let y follow the model in equation (42) and let A be a full-rank linear transformation that sparsifies the signal x . To denoise we:*

1. Compute the coefficients Ay .
2. Apply the hard-block-thresholding operator $\mathcal{B}_\eta : \mathbb{C}^n \rightarrow \mathbb{C}^n$ to the coefficients

$$\mathcal{B}_\eta(v)[j] := \begin{cases} v[j] & \text{if } j \in \mathcal{I}_j \text{ such that } \|v_{\mathcal{I}_j}\|_2 > \eta, \\ 0 & \text{otherwise,} \end{cases} \quad (46)$$

for $1 \leq j \leq N$. The set \mathcal{I}_j is a block of coefficients surrounding index j . The size of each block captures to what extent we expect the coefficients to cluster, and can be adjusted by cross validation. The threshold η can be set according to the standard deviation of Az , or also by cross validation.

3. Compute the estimate by inverting the transform, i.e. setting

$$x_{\text{est}} := L \mathcal{B}_\eta(Ay), \quad (47)$$

where L is a left inverse of A .

Figures 14, 15 and 16 show the results of denoising a speech signal applying block thresholding with blocks of length 5. Figures 17 and 18 show the results of denoising an image by applying block thresholding to its 2D Haar wavelet coefficients, with a block size of 5×5 . As in simple hard thresholding, the threshold is chosen by cross validation on a separate set of signals.

Compared to Wiener filtering, which we will cover later on in the course, STFT or wavelet thresholding tends to preserve high-frequency features better, as it can adapt to the local structure of the noisy signal. Hard thresholding tends to produce artifacts due to particularly large noise coefficients. In audio, these coefficients can be heard as *musical noise*. Block thresholding alleviates this problem, since it is very rare for neighboring noisy coefficients to be large at the same time.

References

- [1] S. Mallat. *A wavelet tour of signal processing: the sparse way*. Academic press, 2008.

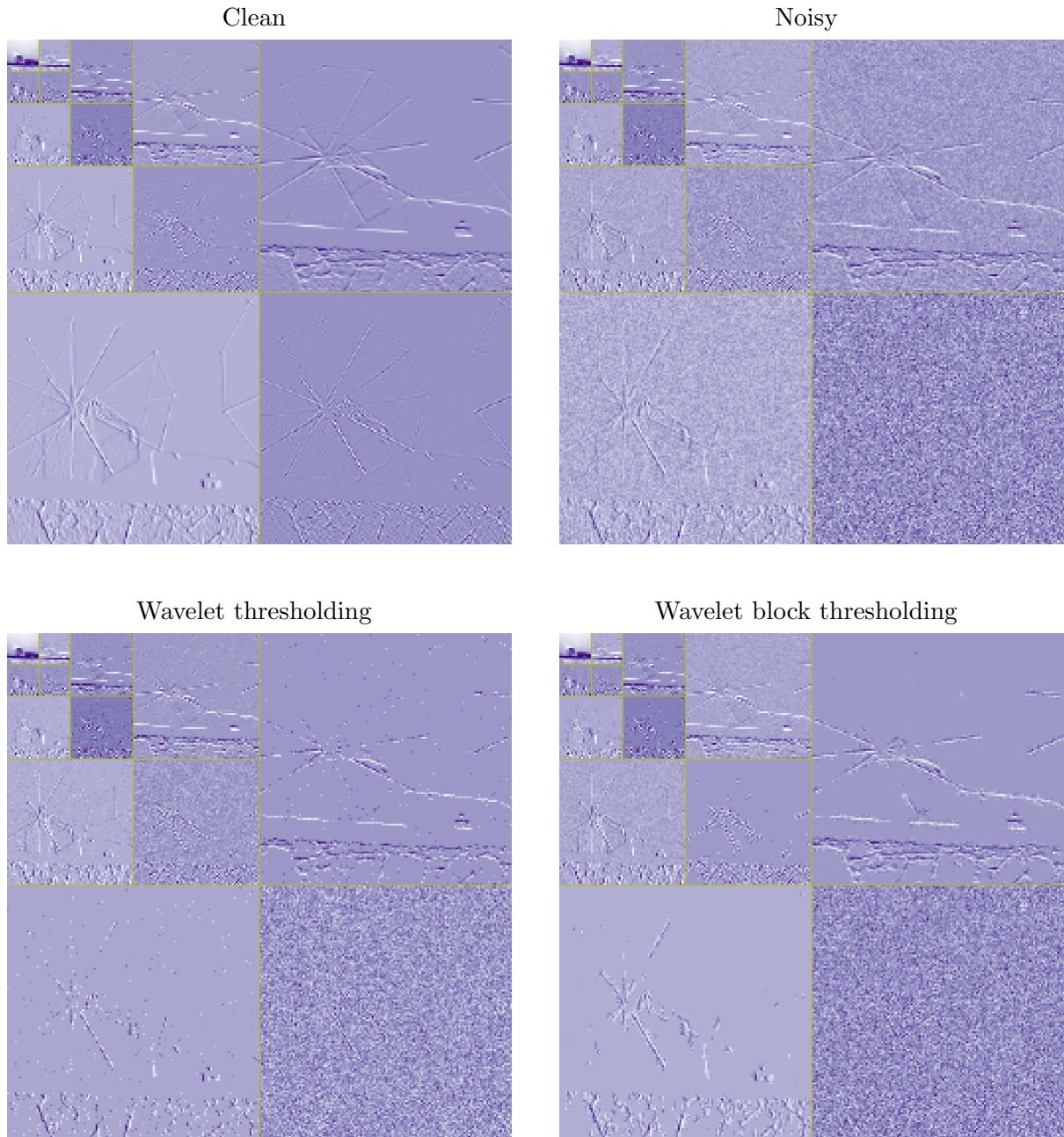


Figure 17: The top row shows the 2D Haar wavelet coefficients of a natural image before and after being corrupted by additive iid Gaussian noise with standard deviation $\sigma = 0.04$. The bottom row shows the result of applying thresholding and block thresholding to the coefficients. The coefficients 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.

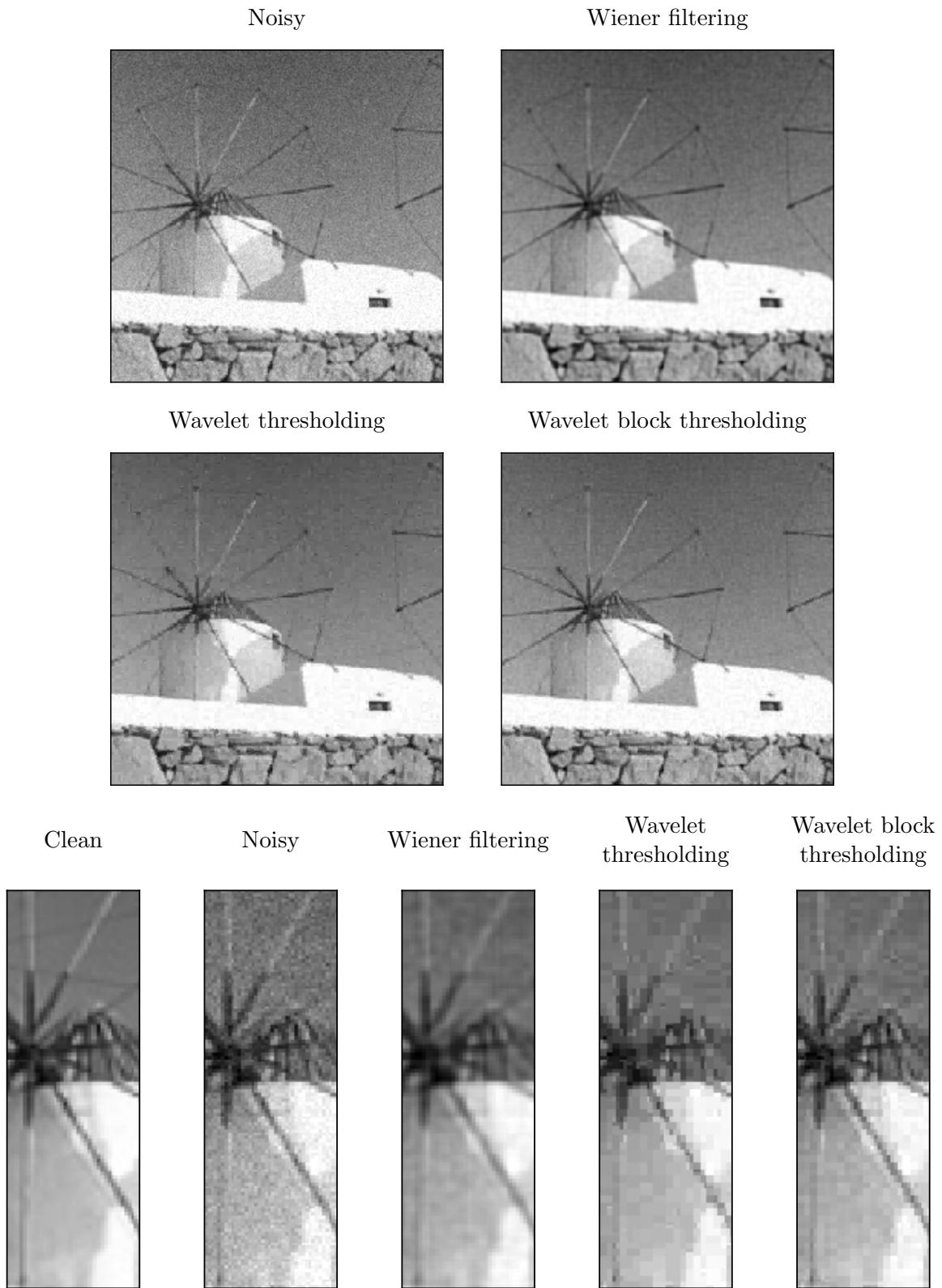


Figure 18: Images corresponding to the coefficients in Figure 17. In addition, the result of applying Wiener filtering to the image is shown for comparison.