Appendix A
Basic Linear Algebra

This book assumes that the student has taken a beginning course in linear algebra at university level. In this appendix we summarize the most important concepts one needs to know from linear algebra. Note that what is included here is very compact, has no exercises, and includes only short proofs, leaving more involved results to other textbooks. The material here should not be considered as a substitute for a linear algebra course: It is important for the student to go through a full such course and do many exercises, in order to get good intuition for these concepts. Recommend books.

Vectors are always written in lowercase boldface ($\boldsymbol{x}$, $\boldsymbol{y}$, etc.), and are always assumed to be column vectors, unless otherwise stated. We will also write column vectors as $\boldsymbol{x} = (x_0, x_1, \ldots, x_n)$, i.e. as a comma-separated list of values, with x_i the components of $\boldsymbol{x}$ (i.e. the components are not in boldface, to distinguish scalars and vectors).

A.1 Matrices

An $m \times n$-matrix is simply a set of mn numbers, stored in m rows and n columns. Matrices are usually written in uppercase (A, B, etc.). We write a_{kn} for the entry in row k and column n of the matrix A. Superscripts are also used to differ between vectors/matrices with the same base name (i.e. $\boldsymbol{x}^{(1)}$, $\boldsymbol{x}^{(2)}$, and $A^{(1)}$, $A^{(2)}$, etc.), so that this does not interfere with the indices of the components.

The zero matrix, denoted $\mathbf{0}$ is the matrix where all entries are zero. A square matrix (i.e. where $m = n$) is said to be diagonal if $a_{kn} = 0$ whenever $k \neq n$. The identity matrix, denoted I, or I_n to make the dimension of the matrix clear, is the diagonal matrix where the entries on the diagonal are 1. If A is a matrix we will denote the transpose of A by A^T. Since vectors are column vectors per default, a row vector will usually be written as $\boldsymbol{x}^T$, with $\boldsymbol{x}$ a column vector. For complex matrices we also define $A^H = (\overline{A})^T$, i.e. as the *conjugate transpose* of A. A real matrix A is said to be *symmetric* if $A = A^T$, and a complex matrix A is said to be *hermitian* if $A = A^H$.

Ø. Ryan, *Linear Algebra, Signal Processing, and Wavelets - A Unified Approach*,
Springer Undergraduate Texts in Mathematics and Technology,
https://doi.org/10.1007/978-3-030-01812-2

If A is an $m \times n$- matrix and $\boldsymbol{x}$ a (column) vector in $\mathbb{R}^n$, then $A\boldsymbol{x}$ is defined as the vector in $\boldsymbol{y} \in \mathbb{R}^m$ with components $y_i = \sum_{j=1}^n a_{ij} x_j$. If the rows of A are denoted $\boldsymbol{a}_1$, $\boldsymbol{a}_2$,...,$\boldsymbol{a}_n$, we can also write

$$A\boldsymbol{x} = x_1 \boldsymbol{a}_1 + x_2 \boldsymbol{a}_2 + \cdots + x_n \boldsymbol{a}_n,$$

i.e. $A\boldsymbol{x}$ can be written as a linear combination of the columns of A. If B is a matrix with n rows,

$$B = \begin{pmatrix} \boldsymbol{b}_1 & \boldsymbol{b}_2 & \cdots & \boldsymbol{b}_k \end{pmatrix},$$

then the product of A and B is defined as

$$AB = \begin{pmatrix} A\boldsymbol{b}_1 & A\boldsymbol{b}_2 & \cdots & A\boldsymbol{b}_k \end{pmatrix}.$$

A square matrix A is said to be *invertible* if there exists a matrix B so that $AB = BA = I$. It is straightforward to show such a B is unique if it exists. B is called the *inverse* of A, and denoted by A^{-1} (so that $A^{-1}A = AA^{-1} = I$). There are many other equivalent conditions for a square matrix to be invertible. One is that $A\boldsymbol{x} = \boldsymbol{0}$ is equivalent to $\boldsymbol{x} = 0$. If the latter is fulfilled we also say that A is *non-singular*. If not, A is said to be *singular*. The terms invertible and non-singular can thus be used interchangeably.

If A is a real non-singular matrix, its inverse is also real. A real matrix A is *orthogonal* if $A^{-1} = A^T$, and a complex matrix A is said to be unitary is $A^{-1} = A^H$. Many matrices constructed in this book are unitary, such as the DFT, the DCT, and some DWT's. A very simple form of unitary matrices are *permutation matrices*, which simply reorders the components in a vector.

A matrix is called sparse if most of the entries in the matrix are zero. Linear systems where the coefficient matrix is sparse can be solved efficiently, and sparse matrices can be multiplied efficiently as well. In this book there are several examples where a matrix can be factored as

$$A = A_1 A_2 \cdots A_n,$$

with the $A_1, A_2, \ldots, A_n$ being sparse. This means that sparse matrix optimizations can be applied for A. This factorization can also be useful when A is a sparse matrix at the beginning, in order to factor a sparse matrix into a product of sparser matrices.

A.2 Block Matrices

If $m_0, \ldots, m_{r-1}$, $n_0, \ldots, n_{s-1}$ are integers, and $A^{(i,j)}$ is an $m_i \times n_j$-matrix for $i = 0, \ldots, r-1$ and $j = 0, \ldots, s-1$, then

$$A = \begin{pmatrix} A^{(0,0)} & A^{(0,1)} & \cdots & A^{(0,s-1)} \\ A^{(1,0)} & A^{(1,1)} & \cdots & A^{(1,s-1)} \\ \vdots & \vdots & \ddots & \vdots \\ A^{(r-1,0)} & A^{(r-1,1)} & \cdots & A^{(r-1,s-1)} \end{pmatrix}$$

will denote the $(m_0 + m_1 + \ldots + m_{r-1}) \times (n_0 + n_1 + \ldots + n_{s-1})$-matrix where the $A^{(i,j)}$ are stacked horizontally and vertically. When A is written in this way it is referred to as an $(r \times s)$ *block matrix*, and the $A^{(i,j)}$ are called the *blocks* of A.

A block matrix is called *block diagonal* if the off-diagonal blocks are zero, i.e.

$$A = \begin{pmatrix} A_0 & 0 & \cdots & 0 \\ 0 & A_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & A_{r-1} \end{pmatrix}.$$

We will also use the notation $\mathrm{diag}(A_0, A_1, \cdots, A_{n-1})$ for block diagonal matrices. The identity matrix is clearly block diagonal, with all diagonal blocks being the identity. The (conjugate) transpose of a block matrix is also a block matrix, and we have that

$$A^T = \begin{pmatrix} (A^{(0,0)})^T & (A^{(1,0)})^T & \cdots & (A^{(r-1,0)})^T \\ (A^{(0,1)})^T & (A^{(1,1)})^T & \cdots & (A^{(r-1,1)})^T \\ \vdots & \vdots & \ddots & \vdots \\ (A^{(0,s-1)})^T & (A^{(1,s-1)})^T & \cdots & (A^{(r-1,s-1)})^T \end{pmatrix}.$$

If A and B are block matrices with blocks $A^{(i,j)}$, $B^{(i,j)}$, respectively, then $C = AB$ is also a block matrix, with blocks

$$C^{(i,j)} = \sum_k A^{(i,k)} B^{(k,j)},$$

as long as A has the same number of horizontal blocks as B has vertically, and as long as each $A^{(i,k)}$ has the same number of columns as $B^{(k,j)}$ has rows.

A.3 Vector Spaces

Given set of objects V (objects are also called *vectors*). V is called a *vector space* if it has an operation for addition, +, and an operation for multiplying with scalars, which satisfy certain properties. First of all $\boldsymbol{u} + \boldsymbol{v}$ must be in V whenever $\boldsymbol{u} \in V$ and $\boldsymbol{V} \in V$, and

- + is commutative (i.e. $\boldsymbol{u} + \boldsymbol{v} = \boldsymbol{v} + \boldsymbol{u}$),
- + is associative (i.e. $(\boldsymbol{u} + \boldsymbol{v}) + \boldsymbol{w} = \boldsymbol{u} + (\boldsymbol{v} + \boldsymbol{w})$),
- there exists a zero vector in V (i.e. a vector $\mathbf{0}$ so that $\boldsymbol{u} + \mathbf{0} = \boldsymbol{u}$ for all $\boldsymbol{u} \in V$).
- For each $\boldsymbol{u}$ in V there is a vector $-\boldsymbol{u} \in V$ so that $\boldsymbol{u} + (-\boldsymbol{u}) = \mathbf{0}$.

Multiplication by scalars, written $c\boldsymbol{u}$ for c a scalar and $\boldsymbol{u} \in V$, is required to satisfy similar properties. Scalars may be required to be either real or complex, in which we talk about *real* and *complex vector spaces.*

We say that vectors $\{\boldsymbol{v}_0, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_{n-1}\}$ in a vector space are *linearly independent* if, whenever $\sum_{i=0}^{n-1} c_i \boldsymbol{v}_i = \mathbf{0}$, we must have that all $c_i = 0$. We will say that a set of vectors $\mathcal{B} = \{\boldsymbol{v}_0, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_{n-1}\}$ from V is a *basis* for V if the vectors are linearly independent, and if any vector in V can be expressed as a linear combination of vectors from $\mathcal{B}$ (we say that $\mathcal{B}$ span V). Any vector in V can then be expressed as a linear combination from vectors in $\mathcal{B}$ in a unique way. Any basis for V has the same number of vectors. If the basis $\mathcal{B}$ for V has n vectors, V is said to have *dimension* n. We also write $\dim(V)$ for the dimension of V. The basis $\{\boldsymbol{e}_0, \boldsymbol{e}_1, \ldots, \boldsymbol{e}_{n-1}\}$ for $\mathbb{R}^n$ is also called the *standard basis* of $\mathbb{R}^n$, and is denoted $\mathcal{E}_n$.

If the $n \times n$-matrix S is non-singular and $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_k$ are linearly independent vectors in $\mathbb{R}^n$, then $S\boldsymbol{x}_1, S\boldsymbol{x}_2, \ldots, S\boldsymbol{x}_k$ are also linearly independent vectors in $\mathbb{R}^n$. If not there would exist $c_1, \ldots, c_k$, not all zero, so that $\sum_{i=1}^k c_i S\boldsymbol{x}_i = \mathbf{0}$. This implies that $S(\sum_{i=1}^k c_i \boldsymbol{x}_i) = \mathbf{0}$, so that $\sum_{i=1}^k c_i \boldsymbol{x}_i = \mathbf{0}$ since S is non-singular. This contradicts that the $\boldsymbol{x}_i$ are linearly independent, so that the $S\boldsymbol{x}_i$ must be linearly independent as well. In other words, if X is the matrix with $\boldsymbol{x}_i$ as columns, X has linearly independent columns if and only if SX has linearly independent columns.

A subset U of a vector space V is called a *subspace* if U also is a vector space. Most of the vector spaces we consider are either subspaces of $\mathbb{R}^N$, matrix spaces, or function spaces. Examples of often encountered subspaces of $\mathbb{R}^N$ are

- the column space $\text{col}(A)$ of the matrix A (i.e. the space spanned by the columns of A),
- the row space $\text{row}(A)$ of the matrix A (i.e. the space spanned by the rows of A), and
- the null space $\text{null}(A)$ of the matrix A (i.e. the space of all vectors $\boldsymbol{x}$ so that $A\boldsymbol{x} = \mathbf{0}$).

It turns out that $\dim(\text{col}(A)) = \dim(\text{row}(A))$. To see why, recall that a general matrix can be brought to row-echelon form through a series of *elementary row operations*. Each such operation does not change the row space of a matrix. Also, each such operation is represented by an non-singular matrix, so that linear independence relations are unchanged after these operations (although the column space itself changes). It follows that $\dim(\text{col}(A)) = \dim(\text{row}(A))$ if and only if this holds for any A in row-echelon form. For a matrix in row-echelon form, however, the dimension of the row- and the column space clearly equals the number of pivot elements, and this proves the result. This common dimension of $\text{col}(A)$ and $\text{row}(A)$ is called the *rank* of A, denoted $\text{rank}(A)$. From what we showed above, the rank of any matrix is unchanged if we multiply with an non-singular matrix to the left or to the right.

Elementary row operations, as mentioned above, can be split into three types: Swapping two rows, multiplying one row by a constant, and adding a multiple of one row to another, i.e. multiplying row j with a constant λ, and add this to row i. Clearly this is the same as computing $E_{i,j,\lambda}A$ where $E_{i,j,\lambda}$ is the matrix $I_m + \lambda \boldsymbol{e}_i \boldsymbol{e}_i^T$, i.e. the matrix which equals the identity matrix, except for an additional entry λ at indices (i,j). The elementary lifting matrices of Chap. 4 combining many such operations into one. As an example it is straightforward to verify that

$$A_\lambda = \prod_{i=0}^{N/2-1} \left(E_{2i,2i-1,\lambda} E_{2i,2i+1,\lambda} \right).$$

As noted in the book, $(A_\lambda)^{-1} = A_{-\lambda}$, which can also be viewed in light of the fact that $(E_{i,j,\lambda})^{-1} = E_{i,j,-\lambda}$.

A.4 Inner Products and Orthogonality

Most vector spaces in this book are inner product spaces. A (real) *inner product* on a vector space is a binary operation, written as $\langle \boldsymbol{u}, \boldsymbol{v} \rangle$, which fulfills the following properties for any vectors $\boldsymbol{u}$, $\boldsymbol{v}$, and $\boldsymbol{w}$:

- $\langle \boldsymbol{u}, \boldsymbol{v} \rangle = \langle \boldsymbol{v}, \boldsymbol{u} \rangle$
- $\langle \boldsymbol{u} + \boldsymbol{v}, \boldsymbol{w} \rangle = \langle \boldsymbol{u}, \boldsymbol{w} \rangle + \langle \boldsymbol{v}, \boldsymbol{w} \rangle$

- $\langle c\boldsymbol{u}, \boldsymbol{v}\rangle = c\langle \boldsymbol{u}, \boldsymbol{v}\rangle$ for any scalar c
- $\langle \boldsymbol{u}, \boldsymbol{u}\rangle \geq 0$, and $\langle \boldsymbol{u}, \boldsymbol{u}\rangle = 0$ if and only if $\boldsymbol{u} = 0$.

$\boldsymbol{u}$ and $\boldsymbol{v}$ are said to be *orthogonal* if $\langle \boldsymbol{u}, \boldsymbol{v}\rangle = \mathbf{0}$. In this book we have seen two important examples of inner product spaces. First of all the Euclidean inner product, which is defined by

$$\langle \boldsymbol{u}, \boldsymbol{v}\rangle = \boldsymbol{v}^T\boldsymbol{u} = \sum_{i=0}^{n-1} u_i v_i \tag{A.1}$$

for any $\boldsymbol{u}$, $\boldsymbol{v}$ in $\mathbb{R}^n$. For functions we have seen examples which are variants of the following form:

$$\langle f, g\rangle = \int f(t)g(t)dt. \tag{A.2}$$

Functions are usually not denoted in boldface, to distinguish them from vectors in $\mathbb{R}^n$. These inner products are real, meaning that it is assumed that the underlying vector space is real, and that $\langle \cdot, \cdots \rangle$ is real-valued. We have also use for *complex inner products*, i.e. complex-valued binary operations $\langle \cdot, \cdots \rangle$ defined on complex vector spaces. A complex inner product is required to satisfy the same four axioms above for real inner products, but the first axiom is replaced by

- $\langle \boldsymbol{u}, \boldsymbol{v}\rangle = \overline{\langle \boldsymbol{v}, \boldsymbol{u}\rangle}$.

This new axiom can be used to prove the property $\langle f, cg\rangle = \bar{c}\langle f, g\rangle$, which is not one of the properties for real inner product spaces. This follows by writing

$$\langle f, cg\rangle = \overline{\langle cg, f\rangle} = \overline{c\langle g, f\rangle} = \bar{c}\overline{\langle g, f\rangle} = \bar{c}\langle f, g\rangle.$$

The inner products above can be generalized to complex inner products by defining

$$\langle \boldsymbol{u}, \boldsymbol{v}\rangle = \boldsymbol{v}^H\boldsymbol{u} = \sum_{i=0}^{n-1} u_i\overline{v_i}, \tag{A.3}$$

and

$$\langle f, g\rangle = \int f(t)\overline{g(t)}dt. \tag{A.4}$$

Any set of mutually orthogonal vectors are also linearly independent. A basis where all basis vectors are mutually orthogonal is called an *orthogonal basis*. If additionally the vectors all have length 1, we say that the basis is *orthonormal*. Regarding the definition of the Euclidean inner product, it is clear that a real/complex square $n \times n$-matrix is orthogonal/unitary if and only if its rows are an orthonormal basis for $\mathbb{R}^n/\mathbb{C}^n$. The same applies for the columns. Also, any unitary matrix preserves inner products, since if $A^H A = I$, and

$$\langle A\boldsymbol{x}, A\boldsymbol{y}\rangle = (A\boldsymbol{y})^H(A\boldsymbol{x}) = \boldsymbol{y}^H A^H A\boldsymbol{x} = \boldsymbol{y}^H\boldsymbol{x} = \langle \boldsymbol{x}, \boldsymbol{y}\rangle.$$

Setting $\boldsymbol{x} = \boldsymbol{y}$ here it follows that unitary matrices preserve norm, i.e. $\|A\boldsymbol{x}\| = \|\boldsymbol{x}\|$.

If $\boldsymbol{x}$ is in a vector space with an orthogonal basis $\mathcal{B} = \{\boldsymbol{v}_k\}_{k=0}^{n-1}$, we can express $\boldsymbol{x}$ as

$$\frac{\langle \boldsymbol{x}, \boldsymbol{v}_0\rangle}{\langle \boldsymbol{v}_0, \boldsymbol{v}_0\rangle}\boldsymbol{v}_0 + \frac{\langle \boldsymbol{x}, \boldsymbol{v}_1\rangle}{\langle \boldsymbol{v}_1, \boldsymbol{v}_1\rangle}\boldsymbol{v}_1 + \cdots + \frac{\langle \boldsymbol{x}, \boldsymbol{v}_{n-1}\rangle}{\langle \boldsymbol{v}_{n-1}, \boldsymbol{v}_{n-1}\rangle}\boldsymbol{v}_{n-1}. \tag{A.5}$$

In other words, the weights in linear combinations are easily found when the basis is orthogonal. This is also called the *orthogonal decomposition theorem*.

By the *projection* of a vector $\boldsymbol{x}$ onto a subspace U we mean the vector $\boldsymbol{y} = \text{proj}_U \boldsymbol{x}$ which minimizes the distance $\|\boldsymbol{y} - \boldsymbol{x}\|$. If $\boldsymbol{v}_i$ is an orthogonal basis for U, we have that $\text{proj}_U \boldsymbol{x}$ can be written by Eq. (A.5).

A.5 Coordinates and Change of Coordinates

If $\mathcal{B} = \{\boldsymbol{b}_0, \boldsymbol{b}_1, \ldots, \boldsymbol{b}_{n-1}\}$ is a basis for a vector space, and $\boldsymbol{x} = \sum_{i=0}^{n-1} x_i \boldsymbol{b}_i$, we say that $(x_0, x_1, \ldots, x_{n-1})$ is the *coordinate vector* of $\boldsymbol{x}$ w.r.t. the basis $\mathcal{B}$. We also write $[\boldsymbol{x}]_\mathcal{B}$ for this coordinate vector.

If $\mathcal{B}$ and $\mathcal{C}$ are two different bases for the same vector space, we can write down the two coordinate vectors $[\boldsymbol{x}]_\mathcal{B}$ and $[\boldsymbol{x}]_\mathcal{C}$. A useful operation is to transform the coordinates in $\mathcal{B}$ to those in $\mathcal{C}$, i.e. apply the transformation which sends $[\boldsymbol{x}]_\mathcal{B}$ to $[\boldsymbol{x}]_\mathcal{C}$. This is a linear transformation, and we will denote the $n \times n$-matrix of this linear transformation by $P_{\mathcal{C}\leftarrow\mathcal{B}}$, and call this the *change of coordinate matrix* from $\mathcal{B}$ to $\mathcal{C}$. In other words, it is required that

$$[\boldsymbol{x}]_\mathcal{C} = P_{\mathcal{C}\leftarrow\mathcal{B}}[\boldsymbol{x}]_\mathcal{B}. \tag{A.6}$$

It is straightforward to show that $P_{\mathcal{C}\leftarrow\mathcal{B}} = (P_{\mathcal{B}\leftarrow\mathcal{C}})^{-1}$, so that matrix inversion can be used to compute the change of coordinate matrix the opposite way. It is also straightforward to show that the columns in the change of coordinate matrix from $\mathcal{B}$ to $\mathcal{C}$ can be obtained by expressing the old basis vectors in terms of the new basis vectors, i.e.

$$P_{\mathcal{C}\leftarrow\mathcal{B}} = \left([\boldsymbol{b}_0]_\mathcal{C} \; [\boldsymbol{b}_1]_\mathcal{C} \cdots [\boldsymbol{b}_{n-1}]_\mathcal{C} \right).$$

In particular, the change of coordinate matrix from $\mathcal{B}$ to the standard basis is

$$P_{\mathcal{E}\leftarrow\mathcal{B}} = \left(\boldsymbol{b}_0 \; \boldsymbol{b}_1 \cdots \boldsymbol{b}_{n-1} \right).$$

If the vectors in $\mathcal{B}$ are orthonormal this matrix is unitary, and we obtain that

$$P_{\mathcal{B}\leftarrow\mathcal{E}} = (P_{\mathcal{E}\leftarrow\mathcal{B}})^{-1} = (P_{\mathcal{E}\leftarrow\mathcal{B}})^H = \begin{pmatrix} (\boldsymbol{b}_0)^H \\ (\boldsymbol{b}_1)^H \\ \cdots \\ (\boldsymbol{b}_{n-1})^H \end{pmatrix}.$$

The DFT and the DCT are such coordinates changes.

The *Gramm matrix* of the basis $\mathcal{B}$ is the matrix with entries being $\langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle$. We will also write $(\langle \mathcal{B}, \mathcal{B} \rangle)$ for this matrix. It is useful to see how the Gramm matrix changes under a change of coordinates. Let S be the change of coordinates from $\mathcal{B}$ to $\mathcal{C}$. Another useful form for this is

$$\begin{pmatrix} \boldsymbol{b}_0 \\ \boldsymbol{b}_1 \\ \vdots \\ \boldsymbol{b}_{n-1} \end{pmatrix} = S^T \begin{pmatrix} \boldsymbol{c}_0 \\ \boldsymbol{c}_1 \\ \vdots \\ \boldsymbol{c}_{n-1} \end{pmatrix},$$

which follows directly since the columns in the change of coordinate matrix is simple the old basis expressed in the new basis. From this it follows that

$$\begin{aligned}\langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle &= \left\langle \sum_{k=0}^{n-1} (S^T)_{ik} \boldsymbol{c}_k, \sum_{l=0}^{n-1} (S^T)_{jl} \boldsymbol{c}_l \right\rangle \\ &= \sum_{k=0}^{n-1} \sum_{l=0}^{n-1} (S^T)_{ik} \langle \boldsymbol{c}_{\boldsymbol{k}}, \boldsymbol{c}_l \rangle S_{lj} = (S^T (\langle \mathcal{C}, \mathcal{C} \rangle) S)_{ij}.\end{aligned}$$

It follows that

$$(\langle \mathcal{B}, \mathcal{B} \rangle) = S^T (\langle \mathcal{C}, \mathcal{C} \rangle) S.$$

More generally when $\mathcal{B}$ and $\mathcal{C}$ are two different bases for the same space, we can define $(\langle \mathcal{B}, \mathcal{C} \rangle)$ as the matrix with entries $\langle \boldsymbol{b}_i, \boldsymbol{c}_j \rangle$. If $S_1 = P_{\mathcal{C}_1 \leftarrow \mathcal{B}_1}$ and $S_2 = P_{\mathcal{C}_2 \leftarrow \mathcal{B}_2}$ are coordinate changes, it is straightforward to generalize the calculation above to show that

$$(\langle \mathcal{B}_1, \mathcal{B}_2 \rangle) = (S_1)^T (\langle \mathcal{C}_1, \mathcal{C}_2 \rangle) S_2.$$

If T is a linear transformation between the spaces V and W, and $\mathcal{B}$ is a basis for V, $\mathcal{C}$ a basis for W, we can consider the operation which sends the coordinates of $\boldsymbol{x} \in V$ in $\mathcal{B}$ to the coordinates of $T\boldsymbol{x} \in W$ in $\mathcal{C}$. This is represented by a matrix, called *the matrix of T relative to the bases $\mathcal{B}$ and $\mathcal{C}$*, and denoted $[T]_{\mathcal{C} \leftarrow \mathcal{B}}$. Thus it is required that

$$[T(\boldsymbol{x})]_{\mathcal{C}} = [T]_{\mathcal{C} \leftarrow \mathcal{B}} [\boldsymbol{x}]_{\mathcal{B}},$$

and clearly

$$[T]_{\mathcal{C} \leftarrow \mathcal{B}} = \left([T(\boldsymbol{b}_0)]_{\mathcal{C}} \; [T(\boldsymbol{b}_1)]_{\mathcal{C}} \; \cdots \; [T(\boldsymbol{b}_{n-1})]_{\mathcal{C}} \right).$$

A.6 Eigenvectors and Eigenvalues

If A is a square matrix, a vector $\boldsymbol{v}$ is called an *eigenvector* if there exists a scalar λ so that $A\boldsymbol{v} = \lambda \boldsymbol{v}$. λ is called the corresponding *eigenvalue*. Clearly a matrix is non-singular if and only if 0 is not an eigenvalue. The concept of eigenvalues and eigenvectors also gives meaning for a linear transformation from a vector space to itself: If T is such a transformation, $\boldsymbol{v}$ is an eigenvector with eigenvalue λ if $T(\boldsymbol{v}) = \lambda \boldsymbol{v}$.

By the *eigenspace* of A corresponding to eigenvalue λ we mean the set of all vectors $\boldsymbol{v}$ so that $A\boldsymbol{v} = \lambda \boldsymbol{v}$. This is a vector space, and may have dimension larger than 1. In basic linear algebra textbooks one often shows that, if A is a real, symmetric matrix,

- the eigenvalues of A are real (this also implies that it has real eigenvectors),
- the eigenspaces of A are orthonormal and together span $\mathbb{R}^n$, so that any vector can be decomposed as a sum of eigenvectors from A.

If we let P be the matrix consisting of the eigenvectors of A, then clearly $AP = DP$, where D is the diagonal matrix with the eigenvalues on the diagonal. Since the eigenvectors are orthonormal, we have that $P^{-1} = P^T$. It follows that, for any symmetric matrix A, we can write

$$A = PDP^T,$$

where P is orthogonal and D is diagonal. We say that A is *orthogonally diagonalizable*. It turns out that a real matrix is symmetric if and only if it is orthogonally diagonalizable.

If A instead is a complex matrix, we say that it is *unitarily diagonalizable* if it can be written as

$$A = PDP^H,$$

For complex matrices there are different results for when a matrix is unitarily diagonalizable. It turns out that a complex matrix A is unitarily diagonalizable if and only if $AA^H = A^H A$. We say then that A is *normal*.

Digital filters as defined in Chap. 3, were shown to be unitarily diagonalizable (having the orthonormal Fourier basis vectors as eigenvectors). It follows that these matrices are normal. This also follows from the fact that filters commute, and since A^H is a filter when A is. Since real filters usually are not symmetric, they are not orthogonally diagonalizable, however. Symmetric filters are of course orthogonally diagonalizable, just as their symmetric restrictions (denoted by S_r), see Sect. 3.5 (S_r was diagonalized by the real and orthogonal DCT matrix).

A matrix A is said to be *diagonalizable* if it can be written on the form

$$A = PDP^{-1},$$

with D diagonal. This is clearly a generalization of the notion of orthogonally/unitarily diagonalizable. Also here the diagonal matrix D contains the eigenvalues of A, and the columns of P are a basis of eigenvectors for A, but they may not be orthonormal: Matrices exist which are diagonalizable, but not orthogonally/unitarily diagonalizable. It i straightforward to show that if A is diagonalizable and it also has orthonormal eigenspaces, then it is also orthogonally/unitarily diagonalizable.

A mapping on the form $T(A) = P^{-1}AP$ with P non-singular is also called a similarity transformation, and the matrices A and $T(A)$ are said to be similar. Similar matrices have the same eigenvalues, since

$$P^{-1}AP - \lambda I = P^{-1}(A - \lambda I)P,$$

so that the characteristic equations for A and $T(A)$ have the same roots. The eigenvectors of A and $T(A)$ are in general different: One can see that if $\boldsymbol{v}$ is an eigenvector for A, then $P^{-1}\boldsymbol{v}$ is an eigenvector for $T(A)$ with the same eigenvalue. Note that A may be orthogonally/unitarily diagonalizable (i.e. on the form $P_0 D P_0^H$), while $T(A)$ is not. If P is unitary, however, $T(A)$ will also be orthogonally/unitarily diagonalizable, since

$$T(A) = P^{-1}AP = P^H P_0 D P_0^H P = (P^H P_0) D (P^H P_0)^H$$

We say that A and $T(A)$ are *unitarily similar*. In Chap. 7 we encountered unitary similarity transformations which used permutation matrices. The permutation grouped each polyphase component together.

A.7 Positive Definite Matrices

A symmetric $n \times n$-matrix A is said to be *positive semidefinite* if $\boldsymbol{x}^H A\boldsymbol{x} \geq 0$ for all $\boldsymbol{x} \in \mathbb{C}^n$. If this inequality is strict for all $\boldsymbol{x} \neq 0$, A is said to be *positive definite*. Any matrix on the form $B^H B$ is positive semidefinite, since $\boldsymbol{x}^H B^H B\boldsymbol{x} = (B\boldsymbol{x})^H (B\boldsymbol{x}) = \|B\boldsymbol{x}\|^2 \geq 0$. If B also has linearly independent columns, $B^H B$ is also positive definite.

Any positive semidefinite matrix has nonnegative eigenvalues, and any positive definite matrix has positive eigenvalues. To show this write $A = UDU^H$ with D diagonal and U unitary. We then have that

$$\boldsymbol{x}^H A\boldsymbol{x} = \boldsymbol{x}^H UDU^H \boldsymbol{x} = \boldsymbol{y}D\boldsymbol{y} = \sum_{i=1}^{n} \lambda_i y_i^2.$$

where $\boldsymbol{y} = U^H \boldsymbol{x}$ is nonzero if and only if $\boldsymbol{x}$ is nonzero. Clearly $\sum_{i=1}^{n} \lambda_i y_i^2$ is nonnegative for all $\boldsymbol{y}$ if and only if all $\lambda_i \geq 0$, and positive for all $\boldsymbol{y} \neq \boldsymbol{0}$ if and only if all $\lambda_i > 0$.

For any positive definite matrix A we have that

$$\|A\boldsymbol{x}\|^2 = \|UDU^H \boldsymbol{x}\|^2 = \|DU^H \boldsymbol{x}\|^2 = \sum_{i=1}^{n} \lambda_i^2 (U^H \boldsymbol{x})_i^2.$$

Also, if the eigenvalues of A are arranged in decreasing order,

$$\begin{aligned}\lambda_n^2 \|\boldsymbol{x}\|^2 = \lambda_n^2 \|U^H \boldsymbol{x}\|^2 = \lambda_n^2 \sum_{i=1}^{n} (U^H \boldsymbol{x})_i^2 \leq \sum_{i=1}^{n} \lambda_i^2 (U^H \boldsymbol{x})_i^2 \\ \leq \lambda_1^2 \sum_{i=1}^{n} (U^H \boldsymbol{x})_i^2 = \lambda_1^2 \|U^H \boldsymbol{x}\|^2 = \lambda_1^2 \|\boldsymbol{x}\|^2.\end{aligned}$$

It follows that

$$\lambda_n \|\boldsymbol{x}\| \leq \|A\boldsymbol{x}\| \leq \lambda_1 \|\boldsymbol{x}\|.$$

Thus, the eigenvalues of a positive definite matrix describe the maximum and minimum increase in the length of a vector when applying A. Clearly the upper and lower bounds are achieved by eigenvectors of A.

A.8 Singular Value Decomposition

In Chap. 6 we encountered frames. The properties of frames were proved using the singular value decomposition, which we now review. Any $m \times n$-matrix A can be written on the form $A = U\Sigma V^H$ where

- U is a unitary $m \times m$-matrix,
- Σ is a diagonal $m \times n$-matrix (in the sense that only the entries $\Sigma_{n,n}$ can be non-zero) with non-negative entries on the diagonal, and in decreasing order, and
- V is a unitary $n \times n$-matrix.

The entries on the diagonal of Σ are called *singular values*, and are denoted by σ_n. $A = U\Sigma V^H$ is called a *singular value decomposition* of A. A singular value decomposition is not unique: Many different U and V may be used in such a decomposition. The matrix Σ, and hence the singular values, are always the same in such a decomposition, however. It turns out that the singular values equal the square roots of the eigenvalues of $A^H A$, and also the square roots of the eigenvalues of AA^H (it turns out that the non-zero eigenvalues of $A^H A$ and AA^H are equal). $A^H A$ and AA^H both have only non-negative eigenvalues since they are positive semidefinite.

Since the U and V in a singular value decomposition are non-singular, we have that $\text{rank}(A) = \text{rank}(\Sigma)$. Since Σ is diagonal, its rank is the number of nonzero entries on the diagonal, so that rank(A) equals the number of positive singular values. Denoting this rank by r we thus have that

$$\sigma_1 \geq \sigma_2 \geq \cdots \sigma_r > \sigma_{r+1} = \sigma_{r+2} = \cdots = \sigma_{\min(m,n)} = 0.$$

In the following we will always denote the rank of A by r. Splitting the singular value decomposition into blocks, A can also be written on the form $A = U_1 \Sigma_1 (V_1)^H$ where

- U is an $m \times r$-matrix with orthonormal columns,
- Σ_1 is the $r \times r$-diagonal matrix $\text{diag}(\sigma_1, \sigma_2, \ldots, \sigma_r)$, and
- V is an $n \times r$-matrix with orthonormal columns.

$A = U_1 \Sigma_1 (V_1)^H$ is called a *singular value factorization* of A. The matrix Σ_1 is non-singular by definition. The matrix with singular value factorization $V_1 (\Sigma_1)^{-1} (U_1)^H$ is called the *generalized inverse* of A. One can prove that this matrix is unique. It is denoted by $A^\dagger$. If A is a square, non-singular matrix, it is straightforward to prove that $A^\dagger = A^{-1}$.

Now, if A has rank n (i.e. the columns are linearly independent, so that $m \geq n$), $A^H A$ is an $n \times n$-matrix where all n eigenvalues are positive, so that it is non-singular. Moreover $V_1 = V$ is unitary in the singular value factorization of A. It follows that

$$A^\dagger A = V_1 (\Sigma_1)^{-1} (U_1)^H = U_1 \Sigma_1 (V_1)^H = V_1 (V_1)^H = I,$$

so that $A^\dagger$ indeed van be considered as an inverse. In this case the following computation also shows that there exists a concrete expression for $A^\dagger$:

$$\begin{aligned}
(A^H A)^{-1} A^H &= (V_1 (\Sigma_1)^T (U_1)^H U_1 \Sigma_1 (V_1)^H)^{-1} V_1 (\Sigma_1)^T (U_1)^H \\
&= (V_1 \Sigma_1^2 (V_1)^H)^{-1} V_1 \Sigma_1 (U_1)^H = V_1 \Sigma_1^{-2} (V_1)^H V_1 \Sigma_1 (U_1)^H \\
&= V_1 \Sigma_1^{-2} \Sigma_1 (U_1)^H = V_1 \Sigma_1^{-1} (U_1)^H = A^\dagger.
\end{aligned}$$

Nomenclature

$(\langle \mathcal{B}_1, \mathcal{B}_2 \rangle)$	Gramm matrix of the bases $\mathcal{B}_1$ and $\mathcal{B}_2$
$[\boldsymbol{x}]_{\mathcal{B}}$	Coordinate vector of $\boldsymbol{x}$ relative to the basis $\mathcal{B}$
$[T]_{\mathcal{C} \leftarrow \mathcal{B}}$	The matrix of T relative to the bases $\mathcal{B}$ and $\mathcal{C}$
$\boldsymbol{x} * \boldsymbol{y}$	Convolution of vectors
$\boldsymbol{x} \circledast \boldsymbol{y}$	Circular convolution of vectors
$\boldsymbol{x}^{(e)}$	Vector of even samples
$\boldsymbol{x}^{(o)}$	Vector of odd samples
$\breve{\boldsymbol{x}}$	Symmetric extension of a vector
$\breve{f}$	Symmetric extension of the function f
$\mathcal{C}_m$	Time-ordering of $(\boldsymbol{\phi}_{m-1}, \boldsymbol{\psi}_{m-1})$
$\mathcal{D}_N$	N-point DCT basis for $\mathbb{R}^N$, i.e. $\{\boldsymbol{d}_0, \boldsymbol{d}_1, \cdots, \boldsymbol{d}_{N-1}\}$
$\mathcal{D}_{N,T}$	Order N real Fourier basis for $V_{N,T}$
$\mathcal{E}_N$	Standard basis for $\mathbb{R}^N$, i.e. $\{\boldsymbol{e}_0, \boldsymbol{e}_1, \cdots, \boldsymbol{e}_{N-1}\}$
$\mathcal{F}_N$	Fourier basis for $\mathbb{R}^N$, i.e. $\{\boldsymbol{\phi}_0, \boldsymbol{\phi}_1, \cdots, \boldsymbol{\phi}_{N-1}\}$
$\mathcal{F}_{N,T}$	Order N complex Fourier basis for $V_{N,T}$
χ_A	Characteristic function for the set A
$\hat{\boldsymbol{x}}$	DFT of the vector $\boldsymbol{x}$
$\hat{f}$	Continuous-time Fourier Transform of f
$\lambda_s(\nu)$	Frequency response of a filter
$\lambda_S(\omega)$	Continuous frequency response of a digital filter
$\langle \boldsymbol{u}, \boldsymbol{v} \rangle$	Inner product
ν	Frequency
ω	Angular frequency
$\oplus$	Direct sum
$\otimes$	Tensor product
$\overline{A}$	Conjugate of a matrix
ϕ	Scaling function
$\phi_{m,n}$	Scaled and translated version of ϕ
σ_i	Singular values of a matrix
DCT_N	$N \times N$-DCT matrix
DFT_N	$N \times N$-DFT matrix
$\mathrm{Supp}(f)$	Support of f

Ø. Ryan, *Linear Algebra, Signal Processing, and Wavelets - A Unified Approach*,
Springer Undergraduate Texts in Mathematics and Technology,
https://doi.org/10.1007/978-3-030-01812-2

$\tilde{\phi}$	Dual scaling function
$\tilde{\psi}$	Dual mother wavelet
$\tilde{V}_m$	Dual resolution space
$\tilde{W}_m$	Dual detail space
$\boldsymbol{\phi}_m$	Basis for V_m
$\boldsymbol{\psi}_m$	Basis for W_m
A^H	Conjugate transpose of a matrix
A^T	Transpose of a matrix
A^{-1}	Inverse of a matrix
$A^\dagger$	Generalized inverse of A
A_λ	Elementary lifting matrix of even type
B_λ	Elementary lifting matrix of odd type
$c_{m,n}$	Coordinates in $\boldsymbol{\phi}_m$
E_d	Filter which delays with d samples
F_N	$N \times N$-Fourier matrix
f_N	N'th order Fourier series of f
f_s	Sampling frequency. Also used for the square wave
f_t	Triangle wave
G	IDWT kernel, or reverse filter bank transform
G_0, G_1	IDWT filter components
H	DWT kernel, or forward filter bank transform
H_0, H_1	DWT filter components
$l(S)$	Length of a filter
N	Number of points in a DFT/DCT
$O(N)$	Order of an algorithm
$P_{\mathcal{C}\leftarrow\mathcal{B}}$	Change of coordinate matrix from $\mathcal{B}$ to $\mathcal{C}$
$S^\leftrightarrow$	Matrix with the columns reversed
S_r	Symmetric restriction of S
T	Period of a function
T_s	Sampling period
V_m	Resolution space
$V_{N,T}$	N'th order Fourier space
W_m	Detail space
$W_m^{(0,1)}$	Resolution m Complementary wavelet space, LH
$W_m^{(1,0)}$	Resolution m Complementary wavelet space, HL
$W_m^{(1,1)}$	Resolution m Complementary wavelet space, HH
$w_{m,n}$	Coordinates in $\boldsymbol{\psi}_m$

References

[1] A. Ambardar, *Digital Signal Processing: A Modern Introduction* (Cengage Learning, Belmont, 2006)

[2] T. Barnwell, An experimental study of sub-band coder design incorporating recursive quadrature filters and optimum ADPCM, in *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP*, pp. 808–811 (1981)

[3] A. Boggess, F.J. Narcowich, *A First Course in Wavelets with Fourier Analysis.* (Prentice Hall, Upper Saddle River, 2001)

[4] O. Bratteli, P. Jorgensen, *Wavelets Through a Looking Glass* (Birkhauser, Boston, 2002)

[5] C.M. Brislawn, Fingerprints go digital. Not. AMS **42**(11), 1278–1283 (1995)

[6] A.A.M.L. Bruekens, A.W.M. van den Enden, New networks for perfect inversion and perfect reconstruction. IEEE J. Sel. Areas Commun. **10**(1), 130–137 (1992)

[7] B.A. Cipra, The best of the 20th century: Editors name top 10 algorithms. SIAM News **33**(4) (2000). http://www.uta.edu/faculty/rcli/TopTen/topten.pdf

[8] A. Cohen, I. Daubechies, Wavelets on the interval and fast wavelet transforms. Appl. Comput. Harmon. Anal. **1**, 54–81 (1993)

[9] A. Cohen, I. Daubechies, J.-C. Feauveau, Biorthogonal bases of compactly supported wavelets. Commun. Pure Appl. Math. **45**(5), 485–560 (1992)

[10] J.W. Cooley, J.W. Tukey, An algorithm for the machine calculation of complex fourier series. Math. Comput. **19**, 297–301 (1965)

[11] A. Croisier, D. Esteban, C. Galand, Perfect channel splitting by use of interpolation/decimation/tree decomposition techniques, in *International Conference on Information Sciences and Systems*, pp. 443–446 (1976)

[12] I. Daubechies, Orthonormal bases of compactly supported wavelets. Commun. Pure Appl. Math. **41**(7), 909–996 (1988)

[13] I. Daubechies, *Ten Lectures on Wavelets.* CBMS-NSF Conference Series in Applied Mathematics (SIAM, Philadelphia, 1992)

[14] I. Daubechies, W. Sweldens, Factoring wavelet transforms into lifting steps. J. Fourier Anal. Appl. **4**(3), 247–269 (1998)

[15] A. Deitmar, *A First Course in Harmonic Analysis*, 2nd edn. (Springer, New York, 2005)

Ø. Ryan, *Linear Algebra, Signal Processing, and Wavelets - A Unified Approach*,
Springer Undergraduate Texts in Mathematics and Technology,
https://doi.org/10.1007/978-3-030-01812-2

[16] P. Duhamel, H. Hollmann, 'split-radix' FFT-algorithm. Electron. Lett. **20**(1), 14–16 (1984)

[17] N. Efford, *Digital Image Processing. A Practical Introduction Using Java* (Addison-Wesley, Boston, 2000)

[18] Y.C. Eldar, *Sampling Theory* (Cambridge University Press, Cambridge, 2015)

[19] FBI, WSQ gray-scale fingerprint image compression specification. Technical report, IAFIS-IC (1993)

[20] S. Foucart, H. Rauhut, *A Mathematical Introduction to Compressive Sensing* (Birkhauser, Basel, 2013)

[21] M.W. Frazier, *An Introduction to Wavelets Through Linear Algebra* (Springer, New York, 1999)

[22] M. Frigo, S.G. Johnson, The design and implementation of FFTW3. Proc. IEEE **93**(2), 216–231 (2005)

[23] T.W. Gamelin, *Complex Analysis* (Springer, New York, 2001)

[24] R.C. Gonzalez, R.E. Woods, S.L. Eddins, *Digital Image Processing Using MATLAB* (Gatesmark Publishing, Knoxville, 2009)

[25] F.J. Harris, On the use of windows for harmonic analysis with the discrete Fourier transform. Proc. IEEE **66**(1), 51–83 (1978)

[26] ISO/IEC, Information technology – coding of moving pictures and associated audio for digital storage media at up to about 1.5 mbit/s. Technical report, ISO/IEC (1993)

[27] ISO/IEC, JPEG2000 part 1 final draft international standard. ISO/IEC FDIS 15444-1. Technical report, ISO/IEC (2000)

[28] S.G. Johnson, M. Frigo, A modified split-radix FFT with fewer arithmetic operations. IEEE Trans. Signal Process. **54**, 1562 (2006)

[29] G. Kaiser, *A Friendly Guide to Wavelets* (Birkhauser, Basel, 1994)

[30] Y. Katznelson, *An Introduction to Harmonic Analysis*, 3rd edn. Cambridge Mathematical Library (Cambridge University Press, Cambridge, 2002)

[31] H.P. Langtangen, *A Primer on Scientific Programming with Python*, 5th edn. (Springer, New York, 2016)

[32] D.C. Lay, *Linear Algebra and Its Applications*, 4th edn. (Addison-Wesley, Boston, 2011)

[33] S.J. Leon, *Linear Algebra with Applications*, 8th edn. (Pearson, Upper Saddle River, 2010)

[34] S. Linge, H.P. Langtangen, *Programming for Computations – MATLAB/Octave* (Springer, New York, 2016)

[35] S. Linge, H.P. Langtangen, *Programming for Computations – Python* (Springer, New York, 2016)

[36] T.D. Lookabaugh, M.G. Perkins, Application of the Princen-Bradley filter bank to speech and image compression. IEEE Trans. Acoust. Speech Signal Process. **38**(11), 1914–1926 (1990)

[37] T. Lyche, *Numerical Linear Algebra and Matrix Factorizations* (Springer, New York, 2018)

[38] J. Ma, G. Plonka, The curvelet transform. IEEE Signal Process. Mag. **27**, 118–133 (2010)

[39] S.G. Mallat, Multiresolution approximations and wavelet orthonormal bases of $L^2(\mathbb{R})$. Trans. Am. Math. Soc. **315**(1), 69–87 (1989)

[40] S. Mallat, *A Wavelet Tour of Signal Processing* (Tapir Academic Press, Boston, 1998)

[41] A. Malthe-Sørenssen, *Elementary Mechanics Using Matlab* (Springer, New York, 2015)
[42] A. Malthe-Sørenssen, *Elementary Mechanics Using Python* (Springer, New York, 2015)
[43] H.S. Malvar, Modulated QMF filter banks with perfect reconstruction. Electron. Lett. **26**(13), 906–907 (1990)
[44] H.S. Malvar, *Signal Processing with Lapped Transforms* (Artech House, Norwood, 1992)
[45] S. Martucci, Symmetric convolution and the discrete sine and cosine transforms. IEEE Trans. Signal Process. **42**, 1038–1051 (1994)
[46] Y. Meyer, Principe d'incertitude, bases hilbertiennes et algebres d'operateurs. Seminaire Bourbaki **662**, 209–223 (1985/1986)
[47] Y. Meyer, Ondelettes et functions splines, in *Seminaire EDP, Ecole Polytecnique*, Paris, France, Dec 1986
[48] Y. Meyer, *Wavelets and Operators* (Cambridge University Press, Cambridge, 1992)
[49] C.D. Meyer, *Matrix Analysis and Applied Linear Algebra* (SIAM, Philadelphia, 2000)
[50] Y. Meyer, R. Coifman, *Wavelets. Calderon-Zygmund and Multilinear Operators* (Cambridge University Press, Cambridge, 1997)
[51] K. Mørken, *Numerical Algorithms and Digital Representation* (UIO, 2013)
[52] P. Noll, MPEG digital audio coding. IEEE Signal Process. Mag. **14**, 59–81 (1997)
[53] P.J. Olver, C. Shakiban, *Applied Linear Algebra* (Pearson, Upper Saddle River, 2006)
[54] A.V. Oppenheim, R.W. Schafer, *Discrete-Time Signal Processing* (Prentice Hall, Upper Saddler River, 1989)
[55] D. Pan, A tutorial on MPEG/audio compression. IEEE Multimedia **2**, 60–74 (1995)
[56] W.B. Pennebaker, J.L. Mitchell, *JPEG Still Image Data Compression Standard* (Van Nostrand Reihnold, New York, 1993)
[57] J.P. Prince, A.B. Bradley, Analysis/synthesis filter bank design based on time domain aliasing cancellation. IEEE Trans. Acoust. Speech Signal Process. **34**(5), 1153–1161 (1986)
[58] J.P. Princen, A.W. Johnson, A.B. Bradley, Subband/transform coding using filter bank designs based on time domain aliasing cancellation, in *IEEE Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, pp. 2161–2164 (1987)
[59] J.G. Proakis, D.G. Manolakis, *Digital Signal Processing. Principles, Algorithms, and Applications*, 5th edn. (Pearson, Upper Saddle River, 2007)
[60] C.M. Rader, Discrete Fourier transforms when the number of data samples is prime. Proc. IEEE **56**, 1107–1108 (1968)
[61] T.A. Ramstad, S.O. Aase, J.H. Husøy, *Subband Compression of Images: Principles and Examples*, vol. 6 (Elsevier, Amsterdam, 1995)
[62] T.D. Rossing, *Handbook of Acoustics* (Springer, New York, 2015)
[63] J.H. Rothweiler, Polyphase quadrature filters – a new subband coding technique, in *ICASSP 83*, Boston, pp. 1280–1283 (1983)
[64] D. Salomon, *Data Compression. The Complete Reference*, 5th edn. (Springer, New York, 2007)
[65] K. Sayood, *Introduction to Data Compression*, 2nd edn. (Morgan Kaufmann, Cambridge, 2000)

[66] C.E. Shannon, Communication in the presence of noise. Proc. Inst. Radio Eng. **37**(1), 10–21 (1949)

[67] M.J.T. Smith, T.P. Barnwell, A new filter bank theory for time-frequency representation. IEEE Trans. Acoust. Speech Signal Process. **35**(3), 314–327 (1987)

[68] P. Stoica, R. Moses, *Spectral Analysis of Signals* (Prentice Hall, Upper Saddler River, 2005)

[69] G. Strang, *Linear Algebra and Its Applications*, 3rd edn. (Brooks/Cole, Belmont, 1988)

[70] G. Strang, T. Nguyen, *Wavelets and Filter Banks* (Cambridge Press, Wellesley, 1996)

[71] W. Sweldens, The lifting scheme: a new philosophy in biorthogonal wavelet constructions, in *Wavelet Applications in Signal and Image Processing III*, pp. 68–79 (1995)

[72] W. Sweldens, The lifting scheme: a custom-design construction of biorthogonal wavelets. Appl. Comput. Harmon. Anal. **3**, 186–200 (1996)

[73] T. Tao, *Analysis II*, 3rd edn. (Springer, New York, 2015)

[74] D.S. Taubman, M.W. Marcellin, *JPEG2000. Image Compression. Fundamentals, Standards and Practice* (Kluwer Academic Publishers, Boston, 2002)

[75] F. Uhlig, *Transform Linear Algebra* (Prentice-Hall, Upper Saddler River, 2002)

[76] M. Vetterli, J. Kovacevic, *Wavelets and Subband Coding* (Prentice Hall, Upper Saddler River, 1995)

[77] M. Vetterli, H.J. Nussbaumer, Simple FFT and DCT algorithms with reduced number of operations. Signal Proc. **6**, 267–278 (1984)

[78] M. Vetterli, E. Kovasevic, V.K. Goyal, *Foundations of Signal Processing* (Cambridge University Press, Cambridge, 2014)

[79] Z. Wang, Fast algorithms for the discrete W transform and for the discrete Fourier transform. IEEE Trans. Acoust. Speech Signal Process. **32**(4), 803–816 (1984)

[80] S. Winograd, On computing the discrete Fourier transform. Math. Comput. **32**, 175–199 (1978)

[81] R. Yavne, An economical method for calculating the discrete Fourier transform. Proc. AFIPS Fall Joint Comput. Conf. **33**, 115–125 (1968)

Index

Ø. Ryan, *Linear Algebra, Signal Processing, and Wavelets - A Unified Approach*,
Springer Undergraduate Texts in Mathematics and Technology,
https://doi.org/10.1007/978-3-030-01812-2