Acoustic Features for Environmental Sound Analysis

Romain Serizel, Victor Bisot, Slim Essid, Gael Richard

Chapter 4: Acoustic Features for Environmental Sound Analysis.

Romain Serizel, Victor Bisot, Slim Essid, Gaël Richard

1 Introduction

The time domain representation of a sound signal, or waveform, is not easy to interpret directly. Most of the time it is nearly impossible, from a waveform, to identify or even localise sound events (unless they occurs at different dynamic range, e.g., a loud noise in a quiet environment) and to discriminate between sound scenes. Therefore, frequency-domain representations and time-frequency domain representations (including multiscale representations) have been used for years providing representations of the sound signals that are more inline with the human perception.

However, these representations are usually too generic and often fail to describe specific content that is present in a sound recording. A lot of work has been devoted to design features that could allow extraction of such specific information, leading to a wide variety of hand-crafted features. One problem with these types of features is that, by design, they are specific to a task and that they usually do not generalise well. They often need to be combined with other features, leading to large feature vectors. During the past years, owing to the increasing availability of medium scale and large scale sound datasets, an alternative approach to feature extraction has

Romain Serizel
Université de Lorraine, LORIA, UMR 7503, Vandœuvre-lès-Nancy, France; Inria, Villers-lès-Nancy, France; CNRS, LORIA, UMR 7503, Vandœuvre-lès-Nancy, France, e-mail: romain.serizel@loria.fr

Victor Bisot
Télécom ParisTech, Université Paris-Saclay, Paris, France, e-mail: victor.bistot@telecom-paristech.fr

Slim Essid
Télécom ParisTech, Université Paris-Saclay, Paris, France, e-mail: slim.essid@telecom-paristech.fr

Gaël Richard
Télécom ParisTech, Université Paris-Saclay, Paris, France, e-mail: gael.richard@telecom-paristech.fr

become popular, the so-called feature learning that has proven competitive with most finely tuned hand-crafted features.

Finally, in both cases, using either feature engineering or feature learning, processing the amount of data that is at hand nowadays can quickly become overwhelming. It is therefore of paramount importance to be able to reduce the size of the dataset in the feature space either by reducing the feature vectors dimensionality or by reducing the amount of feature vectors to process.

The general processing chain to convert a sound signal to a feature vector that can be efficiently exploited by a classifier is described is this chapter. The standard steps are presented sequentially (see also Fig. 1). It is also crucial to design features that are robust to perturbation. Therefore, the possibility to enhance signals or enforce robustness at each step is discussed in the corresponding section when applicable. Finally, the relation to features used for speech and music processing is briefly discussed in Sect. 7 and conclusions are presented in Sect. 8.

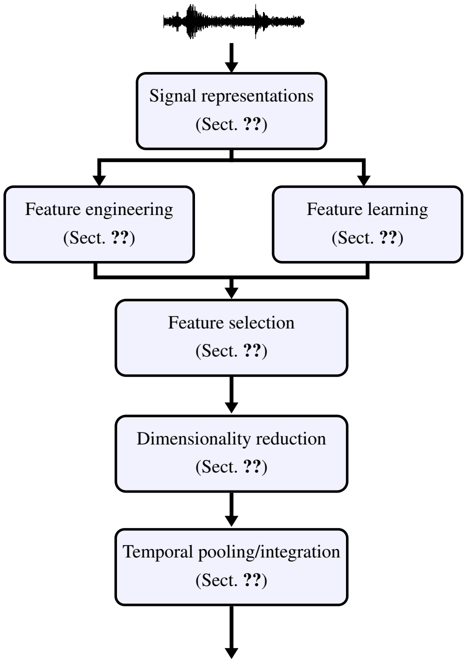


Fig. 1: Standard feature extraction process

2 Signal representations

Over the years, a large amount of work has been devoted to finding appropriate representations that allow extraction of useful information from sound signals. Some of the main classes of sound signals representations are presented in this section.

2.1 *Signal acquisition and preprocessing*

In general terms, sound is the result of a vibration that propagates as waves through a medium such as air or water. Sounds can be recorded under the form of an electric signal $x(t)$ by means of an electroacoustic transducer such as a microphone. This analog signal $x(t)$ can then be converted to a digital signal $x[n]$ and stored on a computer before further analysis. The necessary steps to perform this analog-digital conversion include:

- **A filtering stage**: the analog signal $x(t)$ is low-pass filtered in order to limit its frequency bandwidth in the interval $[0,B]$ where B is the cut-off frequency of the low-pass filter.
- **A sampling stage**: the low-passed analog signal is then digitally sampled at a sampling rate $f_s = 2B$ to avoid the well-known frequency aliasing phenomenon.
- **A quantification stage**: the obtained digital signal is then quantized (e.g. the amplitude of the signal can only take a limited number of predefined values to preserve storage capacity).
- **Optional additional stage**: in some cases, additional preprocessing stages can be performed such as pre-emphasis. This step can be performed under the form of a simple first order finite impulse response (FIR) high-pass filter. Historically, this step was performed on speech signals prior to linear prediction (LP) analysis to cope with its typical $-6dB$ spectral tilt which was shown to be detrimental for LP parameters estimation. In other situations, this step is less justified and is therefore not mandatory.

Typical values for audio CD quality are a sampling rate of $f_s = 44.1\,kHz$ and a quantization on 16 bits per sample leading to a bit rate of 705 600 kbit/s for a single channel audio signal. Higher quality standards include sampling rates of 48, 96 or 192 kHz and quantization on 24 bits.

2.2 *General time-frequency representations*

The sound signals are usually converted to the frequency-domain prior to any analysis. The frequency-domain representation of a signal $x[n]$ on a linear frequency scale can be obtained with the discrete-time Fourier transform (DFT):

$$X(f) = \sum_{n=-\infty}^{\infty} x[n]\mathrm{e}^{-\mathrm{i}2\pi fn} \tag{1}$$

The spectrum $X(f)$ is f_s-periodic in f with f_s the sampling frequency. The frequency $f = \frac{f_s}{2}$ represents the Nyquist-frequency.

The spectrum $X(f)$ can be transformed back to time domain with the inverse discrete time Fourier transform (IDFT):

$$x[n] = \frac{1}{f_s}\int_{-\frac{f_s}{2}}^{\frac{f_s}{2}} X(f)\mathrm{e}^{\mathrm{i}2\pi fn}\mathrm{d}f \tag{2}$$

In practice, the spectrum $X(f)$ is approximated by applying the DFT on a windowed frame of length N of the signal $x[n]$. This is referred to as the short-time Fourier transform (STFT). The f^{th} component of the DFT of the t^{th} frame of $x[n]$ is computed as follows:

$$X(t,f) = \sum_{k=0}^{N-1} w[k]x[tN+k]\mathrm{e}^{\frac{-\mathrm{i}2\pi kf}{N}} \tag{3}$$

where $w[k]$ is a window function (e.g., rectangular, Hamming, Blackman,...) used to attenuate some of the effects of the DFT approximation and to enforce continuity and periodicity at the edge of the frames. Equation (3) is given with a hop between frames equal to the length of the frames (N). This means that there is no overlap between consecutive frames. It is common to choose a hop size that is smaller than the frame length in order to introduce overlap that allows for smoother STFT representation and introduces statistical dependencies between frames.

The t^{th} frame of time domain signal $x[n]$ can be obtained from the discrete spectrum $X(t,f)$ by applying the inverse STFT. Both the STFT and the inverse STFT can be efficiently computed using the fast Fourier transform (FFT) and the inverse fast Fourier transform (IFFT), respectively.

The STFT allows for defining the linear-frequency *spectrogram* which is a 2D representation of a sound where energy in each frequency band is given as a function of time. The spectrogram is then the matrix where each column is the modulus of the DFT of a sound signal frame (see also Fig. 2b).

2.3 Log-frequency and perceptually motivated representations

It is often desirable to search for information in specific frequency bands. This may be achieved by computing energy or energy ratios in predefined frequency bands (see also Fig. 2c). The bands can be equally spaced on the frequency axis, placed according to logarithm or perceptual laws. The number of bands, the shape of the prototype filter and the overlap between bands can also vary greatly.

1. **Critical bands** were introduced by Fletcher [33]. The key idea is that critical bands describe the bandwidth of the auditory filters in the cochlea. Conceptually, this means that two tones within the same critical band will interfere with each other, this is the so-called frequency masking phenomenon. The equivalent rectangular bandwidth scale (ERB) provides a way to compute the central frequency and bandwidth of the rectangular filters approximating the auditory filters [35]:

   $$\mathrm{ERB}(f) = 24.7 \times \left(4.37 \frac{f}{1000} + 1\right) \tag{4}$$

   with f in Hertz. The Bark scale is another scale relying on the concept of critical bands but that was derived from different experiments [95].
2. **Gammatone filters** are linear filters whose impulse response $\mathrm{gamma}[n]$ is composed of a sinusoidal carrier wave (a tone) modulated in amplitude by an envelope that has the same form as a scaled gamma distribution function:

   $$\mathrm{gamma}[n] = an^{\gamma-1}\mathrm{e}^{-2\pi bn}\cos(2\pi f_c n + \Phi), \tag{5}$$

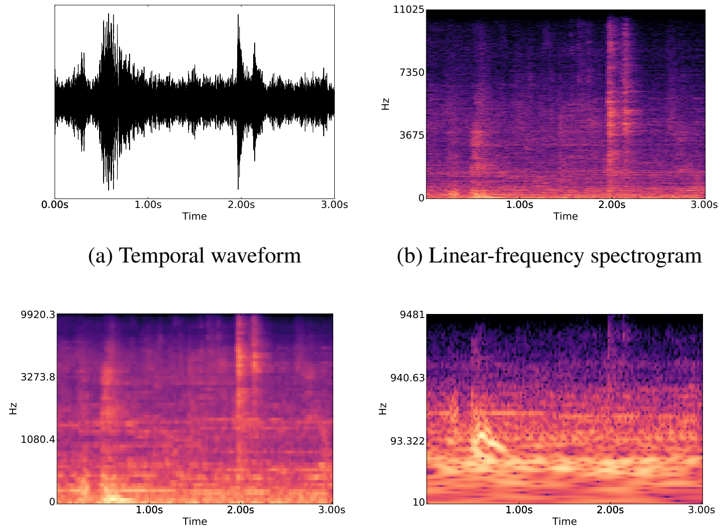
   where a is the amplitude, γ is the filter order, b is a temporal decay coefficient (related to the bandwidth of the filter), f_c the frequency of the carrier (related to centre frequency of the filter) and Φ the phase of the carrier (related to the position of the envelope on the carrier). Similarly to ERB, gammatone filters of order 4 have been shown to provide a good approximation to auditory filters [71].
3. **Mel-scale** corresponds to an approximation of the psychological sensation of heights of a pure sound (e.g., a pure sinusoid) [86]. Several analytical expressions exist [68], a common relation between the mel scale $\mathrm{mel}(f)$ and the Hertz scale f was given by Fant [31]:

   $$\mathrm{mel}(f) = \frac{1000}{\log 2} \log\left(1 + \frac{f}{1000}\right) \tag{6}$$

4. **Constant-Q transform (CQT)** is closely related to DFT. One major difference is that instead of using a frequency scale with constant spacing between frequencies (as in DFT), the frequencies are distributed geometrically [13]. This yields a constant ratio Q between the central frequency of a band f_k and the frequency resolution $f_k - f_{k-1}$, therefore the name CQT. The central frequency for the k^{th} band is the given by:

   $$f_k = f_0 \times 2^{\frac{k}{b}}, \tag{7}$$

   with f_0 the central frequency of the first band and b the number of frequencies per octave (see also Fig. 2d). This transform was originally introduced to map the western musical scale.



(a) Temporal waveform (b) Linear-frequency spectrogram

(c) Mel spectrogram (d) Constant-Q spectrogram

Fig. 2: Different time domain and time-frequency domain representations of a sound signal recorded in a restaurant: at 0.5 s someone is clearing his throat, at 2 s there is some cutlery noises [65].

2.4 Multiscale representations

Multiscale approaches allow for flexible decompositions, representing the sound signal on multiple scales both for time and frequency. Some of the most common approaches are presented below :

1. **Pyramids** are multiscale representations that were originally introduced for image processing [15]. Pyramids are built recursively by applying at each step a convolutive operation (filtering) followed by a down-sampling operation on a signal. This procedure allows to extract information at different resolutions. There are two main-type of pyramids: the so-called Gaussian pyramids (where a low-pass filtering is applied) [15] and the Laplacian pyramids (where a band-pass filtering is applied) [14]. Pyramids with quadratic mirror filters (QMF) [22] have been shown to be closely related to wavelets [60].
2. **Wavelets** are functions that can generally be visualised as a brief oscillation and that should integrate to zero [36, 58]. Given a discrete-time wavelet $\Psi(x)$, it is possible to define a wavelet basis by applying translation and dilatation on the wavelet
$$\Psi_{ab}(x) = \frac{1}{\sqrt{a}}\Psi\left(\frac{x-b}{a}\right) , \tag{8}$$

with $a \in \mathbb{R}^+$ the dilatation factor and $b \in \mathbb{R}$ the translation factor. The translation then allows for covering different time instants while the dilatation of the wavelet enables multiscale analysis [60]. Note that in practice a and b often take their value in a discrete subspace of $\mathbb{R}$, defining so-called discrete wavelets bases.

3. **Scattering transform** builds invariant and stable representations by cascading a wavelet transform, a modulus operation and a low pass filtering operation [59]. Scattering transform can capture non-stationary behavior and can be interpreted as an operation that calculates modulation spectrum coefficients of multiple orders. This approach can enable the modeling of signal dynamics as well as sound textures that are important aspects in the characterization of environmental sounds.

2.5 Discussion

Time-frequency representations such as STFT were designed mainly according to mathematical rules leading for example to linear frequency scales. Human perception studies have shown that we do not perceive sound similarly in each region of the spectrum and that the resolution of the human ear also varies along the frequency axis. Therefore, non-linear frequency scales have been introduced in an attempt to mimic human perception and provide a better way to extract information from sound signals. The frequency scale can be tuned to map the auditory filters (critical bands, ERB, bark scale), to match perceptual behaviour (mel scale) or according to the intrinsic properties of the signal to represent (CQT). In any case, adjusting the granularity of the frequency scale usually allows designing more accurate representations of the signal of interest and can therefore lead to increased robustness. It is also possible to apply standard frequency-domain filtering [29, 39, 93] to time-frequency domain representations in order to attenuate the effects of additive perturbations.

Perceptually motivated time-frequency representation often constitute an important part of sound scene and event analysis systems. They serve, either as a way to visually observe the time-frequency content of the sound scene, or as an input representation to more complex classification systems. Therefore, in many cases, their computation is one the first steps for applying some of the feature engineering or feature learning techniques presented in Sects. 3 and 4. Extracting representations based on mel or gammatone filter-banks can be necessary to compute cepstral features (see Sect. 3.3), which are widely popular in the field [73, 90]. Other representations such as the CQT are often used to build time-frequency images from which image-based features are extracted [10, 79, 94]. Such representations are also considered as inputs to feature learning techniques such as nonnegative matrix factorisation [6, 11, 21], or can be directly used as features for deep neural network based systems [70, 75].

Yet, in these approaches there is only one fixed frequency scale that is non-linear and the time scale remains linear. As sound signals contain information at different time and frequency scales, parts of the signal might be overlooked with these representations. Some works based on variants of the scattering transform proved the usefulness of multiscale representations to perform sound event classification in real life conditions [56, 82].

3 Feature engineering

Similarly to other sound processing tasks, feature extraction for sound scene and event analysis has often relied on so-called feature engineering. This is the art of carefully crafting ad-hoc features from low level representations heavily relying on expert knowledge about class invariances. Some of the most common feature classes are presented in this section (see also Fig. 3).

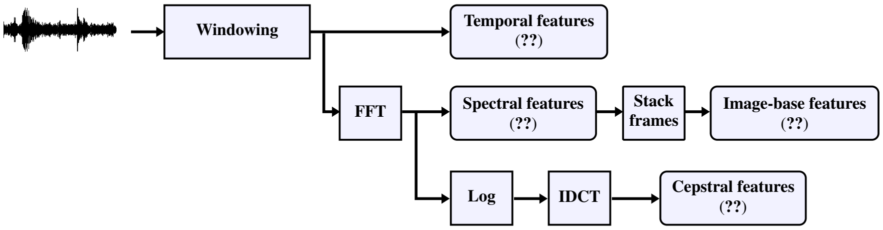


Fig. 3: Feature engineering process.

3.1 Temporal features

These features are computed directly on the temporal waveform and are therefore usually rather straightforward to compute. Some of the most common temporal features are described below.

1. **Time domain Envelope** can be seen as the boundary within which the signal is contained. A simple implementation relies on the computation of the root mean square of the mean energy of the signal $x[n]$ within a frame t of size N spanning over the time indexes $n \in \{n_t, n_t+1, \dots n_t+N\}$:

$$e(t) = \sqrt{\frac{1}{N} \sum_{n=n_t}^{n_t+N} x[n]^2} \, .$$

   It is a reliable indicator for silence detection.

2. **Zero crossing rate (ZCR)** is given by the number of time the signal amplitude crosses the zero value. For a frame t of size N, it is given by:

$$z_{\mathrm{cr}}(t) = \frac{1}{2}\sum_{n=n_t}^{n_t+N} |\mathrm{sgn}(x[n]) - \mathrm{sgn}(x[n-1])| \ , \tag{9}$$

   where $sign(x[n])$ returns the sign of the signal amplitude $x[n]$. It is a very popular feature since it can, in a simple manner, discriminate periodic signals (small ZCR values) from signals corrupted by noises that are random to a certain degree (high ZCR values).
3. **Temporal waveform moments** allow the representation of different characteristics of the shape of the time domain waveform. They are defined from the first four central moments and include the following characteristics:

   - centre of gravity of the waveform: *temporal centroid*,
   - spread around the mean value: *temporal width*,
   - waveform asymmetry around its mean: *temporal asymmetry*,
   - overall flatness of the time domain waveform: *temporal flatness*.

   Note that these moments can also be computed on the spectrum (see below).
4. **Autocorrelation coefficients** can be interpreted as the signal spectral distribution in the time domain. In practice, it is common to only consider the first K coefficients which can be obtained as:

$$R(k) = \frac{\sum_{n=0}^{N-k-1} x[n]x[n+k]}{\sqrt{\sum_{n=0}^{N-k-1} x^2[n]}\sqrt{\sum_{n=0}^{N-k-1} x^2[n+k]}}$$

3.2 Spectral shape features

Studies on the perception of sound widely rely on the frequency content of sound signals. Therefore, it is a natural choice to derive features from frequency representations of a signal, for example its spectrogram. Some of the most common spectral features are described below.

1. **Energy** is one of the most straight forward yet important sprectral feature. This feature can be computed directly as a sum of the squared amplitude components $|X(t,f)|$ in the band. It is also common to compute the log-energy in a band.
2. **Spectral envelope** is conceptually similar to time domain envelope but in the frequency domain. It can be seen as the boundary within which the spectrum of a signal is contained. The spectral envelope can be approximated for example using linear predictive coding (LPC) [69].
3. **Spectral moments** describe some of the main spectral shape characteristics. They include the spectral centroid, the spectral width, spectral asymmetry and spectral flatness. They are computed in the same way as the temporal waveform

moments features by replacing the waveform signal $x[n]$ by the Fourier frequency components $X(t,f)$ of the signal.

4. **Amplitude spectral flatness** is an alternative to the spectral flatness feature. It is computed as the ratio between the geometric and the arithmetic means of the spectral amplitude (globally or in several frequency bands).
5. **Spectral slope** measures the average rate of spectral decrease with frequency (more details can be obtained by Peeters [72]).
6. **Spectral roll-off** is defined as the frequency under which a predefined percentage (typically between 85% and 99%) of the total spectral energy is present.
7. **Spectral flux** characterises the dynamic variation of the spectral information. It is either computed as the derivative of the amplitude spectrum or as the normalised correlation between successive amplitude spectra.
8. **Spectral irregularity features** aims at a finer information description linked to the sound partials (e.g. individual frequency components of a sound). Several approaches have been proposed to estimate these features [72].

In sound scene and event analysis, the temporal and spectral shape features are rarely used separately. In fact, they are mostly simple features designed to model specific aspects of the signal and thus are most often combined with several other features. The log mel energy features are a notable exception, they are powerful enough to be used on their own as input for classification or feature learning. Only a few earlier studies have compared their individual effectiveness for the task [17, 73]. Instead, the temporal and spectral shape features are more often considered and evaluated together as one set of features sometimes referred to as low-level features.

3.3 Cepstral features

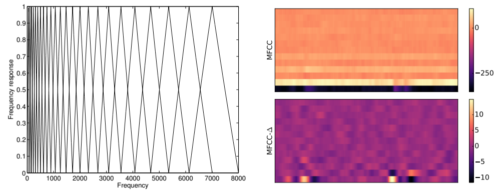
Cepstral features allows the decomposition of the signal according to the so-called source-filter model widely used to model speech production. The signal is then decomposed into a carrier (the source, for speech it can be the glottal excitation) and a modulation (the filter, for speech it includes the vocal tract and the position of the tongue).

1. **Mel-frequency cepstral coefficients (MFCC)** are the most common cepstral coefficients [23]. They are obtained as the inverse discrete cosine transform of the log energy in mel frequency bands:

$$\mathrm{mfcc}(t,c) = \sqrt{\frac{2}{M_{\mathrm{mfcc}}}} \sum_{m=1}^{M_{\mathrm{mfcc}}} \log\left(\tilde{X}_m(t)\right) \cos\left(\frac{c\left(m-\frac{1}{2}\right)\pi}{M_{\mathrm{mfcc}}}\right), \tag{10}$$

where M_{mfcc} is the number of mel frequency bands, m the frequency band index, $\tilde{X}_m(t)$ is the energy in the m^{th} mel frequency band and c is the index of the cepstrum coefficient ($c \in \{1,2,\dots,M_{\mathrm{mfcc}}\}$) (see also Fig. 4b).

In practice, a common implementation uses a triangular filter bank where each filter is spaced according to a mel frequency scale (6) (see also Fig. 4a). The energy coefficients $\tilde{X}_m(t)$ in the band m are obtained as a weighted sum of the spectral amplitude components $|X(t,f)|$ (where the weights are given according to the amplitude value of the corresponding triangular filter). The number M_{mfcc} of filters typically varies between 12 to 30 for a bandwidth of 16 kHz. MFCC are widely used for speech processing but they are also among the most popular features for sound scene analysis [73].



(a) (b)

Fig. 4: Mel filterbank (a) and MFCC decomposition (b)

2. **Alternative cepstral decompostions** can be obtained similarly to MFCC from other frequency-domain representations. This had led to the introduction of features such as the linear prediction cepstral coefficients (LPCC) based on LPC coefficients, the gammatone feature cepstral coefficients (GFCC) or constant-Q cepstral coefficients (CQCC). None of these features are as popular as the MFCC but GFCC for example have been applied to sound scene analysis [74, 90].

3.4 Perceptually motivated features

Studies on human perception have allowed for a better understanding of the human hearing process. Some results from theses studies (such as results on auditory filters) have been exploited in feature engineering and led to widely used features such as MFCC. However, there is still a large variety of perceptual properties that could be exploited in feature extraction (see [80] for a list of common perceptually motivated features for audio classification). To illustrate this category, three perceptual features are described below:

1. **Loudness** (measured in sones) is the subjective impression of the intensity of a sound in such a way that a doubling in sones corresponds to a doubling of loudness. It is commonly obtained as the integration of the specific loudness $L(m)$ over all ERB bands:

$$L = \sum_{m=1}^{M_{\mathrm{ERB}}} L(m) \; , \tag{11}$$

with M_{ERB} the number of ERB bands. The loudness in each band can be approximated [72] by :

$$L(m) = \tilde{X}_m^{0.23} \tag{12}$$

where $\tilde{X}_m$ is the energy of the signal in the m^{th} band (see also (4)).

2. **Sharpness** can be interpreted as a spectral centroid based on psychoacoustic principle. It is commonly estimated as a weighted centroid of specific loudness [72].
3. **Perceptual spread** is a measure of the timbral width of a given sound. It is computed as the relative difference between the largest specific loudness and the total loudness:

$$S_p = \left(\frac{L - max_m(L(m))}{L}\right)^2 \tag{13}$$

3.5 Spectrogram image-based features

Features can also be extracted from the time-frequency representation of a sound scene. Spectrogram image-based features rely on techniques inspired by computer vision to characterise the shape, texture and evolution of the time-frequency content in a sound scene. Such features have proven to be competitive with more traditional audio features on some sound scene classification tasks [79, 47].

1. **Histogram of oriented gradients (HOG)** are image-based features used in computer vision to perform shape detection in images. They are computed from a spectrogram image of a sound scene with the goal of capturing relevant time-frequency structures for characterising sound scenes and events [79]. They are usually extracted by computing a gradient image containing the gradients of each pixel in a spectrogram image. Each pixel of the gradient image represents the direction of the change in intensity in the original image. After separating the image in non overlapping cells, a histogram of the gradient orientations for each pixel is computed in each cell. Variations for the HOG features include the choice of the cells, the normalisation of the histograms and the number of orientations.
2. **Subband power distribution (SPD)** rely on a transformation of a time-frequency image into a two dimensional representation of frequency against spectral power [24]. They are computed by estimating the spectral distribution in each subbands of a spectrogram. In practice the distributions are estimated by extracting a histogram of the pixel values in each subband. The SPD image can either directly be used as features [24] or as an intermediate representation for extracting other image-based features [10].
3. **Local binary pattern (LBP)** analysis is a feature extraction technique used in image recognition to characterise textures in an image. The LBP features are binary vectors associated with each pixel in an image. They are build by comparing the value of a given pixel to others in a fixed neighborhood. For example, local binary patterns can formed by comparing a given pixel to its eight neighbors,

leading to a vector of size eight filled by attributing a value of one to neighbor pixels that have a value above the center pixel and zero to the others. Similarly to the HOG features, the final LBP features are often obtained by computing the distribution of the different local binary patterns in regions of the image. LBP have been applied sound scene analysis in order to capture the texture and geometrical properties of a scene's spectrogram [4, 47].

3.6 Discussion

"Hand-crafted" features are generally very successful for sound analysis tasks but, very few works in sound scene and event analysis focused on creating features adapted to the specificity of the problem. Instead, a more common approach is to select and adapt features initially introduced for other tasks. A now well established example of this trend is the popularity of MFCC features in sound scene and event analysis systems. Although many studies have proved the superiority of other "hand-crafted" features for the task, many systems limit themselves to the use of MFCCs while mostly focusing on the classification and detection stage.

One advantage of this approach is that it allows to re-use the work done on MFCC. For example time domain and frequency-domain filtering [29, 39, 93] to enforce robustness to additive perturbations or cepstral mean normalisation [52] to attenuate the effects of convolutive perturbations. One of the main drawbacks of feature engineering is that it relies on transformations that are defined beforehand and regardless of some particularities of the signals observed at runtime (recording conditions, recording devices. . .).

4 Feature learning

Representation learning techniques have recently proven superior to manually designed features in many classification and other sound analysis tasks. Indeed more and more datasets of significant size have become available that can be used to develop feature learning techniques. Developments in nonnegative matrix factorisation [53], sparse representation learning [40], dictionary learning [57] and deep learning [8] are manifestations of this trend. This approach allows for extracting features that reflect the underlying structure of the data considered in a particular task, providing high level representations that can generalise, to some extent, to data configurations unseen during the training phase.

The potential of feature learning techniques is particularly clear for sound scene event analysis. In fact, real life sound events can be of very different nature resulting in a wide variety of possible time-frequency structures present in a sound scene. Moreover, for tasks like sound scene or event classification, only parts of the information is relevant to discriminate the different target sound object classes. The usefulness of feature learning has already been demonstrated on many scene and event classification datasets. For example, works relied on clustering [83], bag-of-features [76, 94] or nonnegative matrix factorisation [5, 11, 64] techniques in order to learn more discriminative representations of sound scenes and events.

4.1 Deep learning for feature extraction

During the past decade, advances in terms of training algorithms [42, 92] and computing power have lead to the generalisation of the use of deep learning techniques [7] that are now the state-of-the-art in many audio applications. Besides their most common application in pattern classification (see also Chap. 5) deep learning techniques such as deep neural networks (DNN) (Chap. 5, Sect. 4.2), convolutional neural networks (CNN) (Chap. 5, Sect. 4.3), recurrent neural networks (RNN) (Chap. 5, Sect. 4.4) can be applied to learn features. A particular type of network architecture that is often used in feature learning are the so called bottleneck networks (BN) that contain a hidden layer which size is smaller than other hidden layers. There are then two main different strategies than can be applied to learn features with deep learning:

1. **Supervised learning:** When annotated data is available it is often desired to train the network in a supervised manner in order to learn features that are discriminative between the target classes. At run time, in the case of DNN, the last hidden layer is used to extract features [41] while in BN it is the bottleneck layer [37] that provides the features.
2. **Unsupervised learning:** With the increasing amount of data at hands it is often the case that at least part of the data available is not annotated. In this case, feature learning will have to rely on unsupervised techniques in order to extract intrinsic properties of the sound signals. Deep networks can then be trained with restricted Boltzmann machine [42] or stacked auto-encoder [92]. In the latter approach, the network is built gradually by combining denoising autoencoders [91]. An autoencoder is a neural network with one hidden layers whose targets are low level representations of the sound signal. The input of the autoencoder is generally obtained from a (artificially) degraded version of the sound signal. During the training phase the autoencoder then aims at learning how to reconstruct a clean signal from a noisy signal. At run-time, the feature extraction is generally performed similarly as in the supervised case.

More technical details about deep learning in general, network topologies and learning algorithms in particular can be found in Chap. 5, Sect. 4.

4.2 Matrix factorisation techniques

Matrix factorisation (MF) techniques are non-supervised data decomposition techniques, akin to latent variable analysis. In sound analysis applications it generally consists in explaining a set of frequency representations for T frames $\{\mathbf{v}_1, \cdots, \mathbf{v}_T\}$, as linear combinations of *basis vectors*, also called *dictionary elements*, *atoms*, *elementary patterns*, or *topics*. This is accomplished by determining an approximation of the matrix $\mathbf{V} = \left[v_{f,t}\right]$ assembled by stacking the observations column-wise, under the form:

$$\mathbf{V} \approx \hat{\mathbf{V}} = \mathbf{WH} \tag{14}$$

where $\mathbf{W} = \left[w_{fk}\right]$ is a $F \times K$-matrix whose columns $\mathbf{w}_k$ are the basis vectors; and $\mathbf{H} = [h_{kt}]$ is a $K \times T$-matrix whose elements are the so-called *activation coefficients*, *encodings* or *regressors*.

In the following, the t^{th} column of $\mathbf{H}$ will be denoted by $\mathbf{h}_t$, whereas $\mathbf{h}_{k:}$ will denote its k^{th} row relating to the sequence of activations of basis vector $\mathbf{w}_k$.

Generally, MF has been employed as a means of addressing diverse machine learning or signal processing tasks, including clustering, topics recovery, temporal segmentation and structuring, source separation or feature learning. Here, we focus on the latter usage, which have proven effective in sound scene and event analysis applications [12].

In such scenarios, the observations correspond to an appropriate low-level representation, usually a variant of time-frequency representations (described in Sect. 2.2), e.g., mel-spectra. These time-frequency representations are analysed by MF, in the training stage, in order to obtain a dictionary $\mathbf{W}$ to be used to decompose both training examples and new test observations $\mathbf{v}_t$, yielding feature vectors $\mathbf{h}_t$, to be processed by a classifier.

Various data decomposition methods may actually be described with the matrix factorisation formalism, which optimises different criteria, notably principal component analysis (PCA) [43] (see Sect. 5.1), independent component analysis [20] and nonnegative matrix factorisation (NMF) [53]. The latter has been found to be a particularly effective feature learning approach in the context of sound scene analysis [11, 21] and event classification [64, 6]. Hence it is briefly described hereafter.

The technique, which has actually been known for more than 30 years, was popularised by Lee et al. [53] who demonstrated its ability to learn “the parts of objects” through an application to face image decomposition. This tendency to decompose data in a “natural” way, is due to the constraint imposed to both the dictionary and the activation, that is all coefficients of $\mathbf{W}$ and $\mathbf{H}$ are constrained to be nonnegative.

$\mathbf{W}$ and $\mathbf{H}$ are obtained by minimising a measure of fit $D(\mathbf{V}|\mathbf{WH})$, while imposing the nonnegativity of $\mathbf{W}$ and $\mathbf{H}$, which is approached as a constrained optimisation problem. Unfortunately, this problem is not jointly convex in $(\mathbf{W}, \mathbf{H})$, and hence admits numerous local and global minima. This is one of the principal reasons that have led researchers to consider imposing different types of additional constraints on $\mathbf{W}$ or $\mathbf{H}$, based on prior knowledge available when handling a particular application.

In many cases, constraints have been expressed through the choice of a form of regularised objective function, such as:

$$C(\mathbf{W},\mathbf{H}) = D(\mathbf{V}|\mathbf{WH}) + \lambda S(\mathbf{H}) + \eta R(\mathbf{W}) \tag{15}$$

where $S(\mathbf{H})$ and $R(\mathbf{W})$ are constraints on the coefficients of **H** and **W**, respectively. Different types of constraints have been imagined, notably *sparsity constraints*—possibly *group sparsity*—on either **W** or **H**, which is usually translated into sparsity-inducing penalties (e.g., [26, 44, 87]). Such strategies are quite natural in a feature learning context where they are akin to sparse coding.

Fortunately, for many choices of measure of fit $D(\mathbf{V}|\mathbf{WH})$ and penalties $S(\mathbf{H})$ and $R(\mathbf{W})$, the objective function C(**W**,**H**) is separately convex w.r.t **W** for **H** fixed and *vice-versa*. Consequently, most methods aiming to solve the minimisation problem, adopt a *block-coordinate descent* approach whereby update rules are alternately applied to iterates of **W** and **H** [51].

The choice of an appropriate measure-of-fit function $D(\mathbf{V}|\mathbf{WH})$ is of course crucial. It is usually chosen to be a *separable matrix divergence*, taking the form:

$$D(\mathbf{V}|\hat{\mathbf{V}}) = \sum_{f=1}^{K}\sum_{t=1}^{T} d\left(v_{f,t}|\hat{v}_{ft}\right) \tag{16}$$

where $d(x|y)$ is a scalar divergence. A function $d(x|y)$ is said to be a divergence if it is *i)* continuous over x and y; *ii)* $d(x|y) \geq 0 \; \forall x, y \geq 0$; and *iii)* $d(x|y) = 0$ if and only if $x = y$.

Many variants have been considered in previous works including the β-divergence [28], the general Bregman divergences [25], the α-divergences [18] and Csiszar's divergences [19], to mention a few of them. When considering sound signals, it is common to exploit the β-divergence, focusing on particular cases which have proven sufficiently well-adapted to our applications. Special cases of the β-divergence yield popular cost functions, namely: the Itakura-Saito (IS) divergence [32] ($\beta = 0$), Kullback-Leibler (KL) divergence ($\beta = 1$) and the ℓ_2-norm or squared Euclidian distance ($\beta = 2$).

4.3 Discussion

Feature learning techniques have seen an increase in popularity for sound scene and event analysis applications in the last few years. They mainly aim at addressing the general limitations of hand-crafted features mentioned in Sect. 3.6 and have proven to be viable alternatives. Techniques such as NMF have shown, on multiple occasions, to provide better representations than most feature engineering-based methods. For example, NMF allowed to reach improved performance on sound scene and event classification problems, either by considering the dictionaries learned on individual sounds as features [16] or by keeping the projections on a common dictionary representing the full training data as features [11]. Further improvements have been attained by using sparse and convolutive variants of NMF [11, 21, 48]. Another commonly used dictionary learning technique is probabilistic latent component analysis (a probabilistic equivalent of NMF), which has mostly been applied in its temporally constrained shift-invariant version [5, 6]. Other successful unsupervised feature learning approaches include the use of spherical K-means [83], bag-of-features [76, 94] for classifying sound scenes and events. Interested reader is referred to the corresponding references for further information about these feature learning techniques.

Another trend in sound scene and event analysis has been to introduce supervised variants of some of the feature learning techniques mentioned above. For classification problems, supervised feature learning mainly aims at incorporating prior knowledge about the class labels during the feature learning stage in order to learn more discriminant representations of the data. Once again, several supervised extensions of NMF have been proposed. For acoustic event detection, some works incorporated the sequence of labels in the data before decomposing with NMF or convolutive NMF [48, 64]. Moreover, for sound scene classification, supervision has been introduced to NMF either by learning a nonnegative dictionary and a classifier in a joint optimisation problem [12] or by constraining each dictionary elements to represent only one sound label [77].

5 Dimensionality reduction and feature selection

A large number of potentially useful features can be considered in the design of sound scene or event classification systems. Though it is sometimes practicable to use all those features for the classification, it may be sub-optimal to do so, since many of them may be redundant or, even worse, noisy owing to non robust extraction procedures. Thus, feature selection or compression (by transformation) become inevitable in order to reduce the complexity of the problem—by reducing its dimensionality—and to retain only the information that is relevant in discriminating the target classes.

5.1 Dimensionality reduction

A common approach to cope with the potentially large dimensionality of the feature space is to use transformation techniques such as PCA, linear discriminant analysis (LDA), or more recent approaches such as so-called bottleneck DNN. Here, we focus on the popular PCA technique.

PCA, also known as the Karhunen-Loeve transform, computes low-dimensional linear approximations $\hat{\mathbf{v}}$ of the original data points $\mathbf{v}$ in the least-squares sense, that is by seeking a transformation matrix $\mathbf{U}^*$ such that $\mathbf{U}^* = \arg\min_{\mathbf{U}} ||\hat{\mathbf{v}} - \mathbf{v}||^2$, with $\hat{\mathbf{v}} = \mathbf{U}\mathbf{U}^\mathsf{T}\mathbf{v}$ and $rank(\mathbf{U}) < F$. This can be viewed as a projection of the initial data $\mathbf{v}$ on the new coordinate axes for which the variances of $\mathbf{v}$ on these axes are maximized.

The method is actually a special case of matrix factorisation, previously presented, where $\mathbf{W} = \mathbf{U}$ and $\mathbf{H} = \mathbf{U}^\mathsf{T}\mathbf{V}$. Thus, the procedure can be viewed as a projection of the initial data points $\mathbf{v}$ on new coordinate axes, called *principal components*. It is worth noting that other matrix factorisation variants (presented in Sect. 4.2) can be used for dimensionality reduction, as long as $K < F$, merely using the activation vectors $\mathbf{h}_t$ as low-dimensional representatives of the original $\mathbf{v}_t$ data points.

Solving the PCA least-squares problem is shown to be equivalent to computing an eigen value decomposition (EVD) of the covariance matrix $\mathbf{R}_{\mathbf{vv}}$ of the data and taking $\mathbf{U}$ to be the K dominant eigenvectors of this decomposition. This yields the best K-dimensional approximation of the original data in the least-squares sense. It can then be easily verified that the covariance matrix of the transformed data is diagonal, hence the components of the transformed data $\hat{\mathbf{v}}$ are uncorrelated, and the first few components (the so-called principal components) capture most of the variance of the original data $\mathbf{x}$. The interested reader is referred to Murphy [67] for more details about the method.

5.2 Feature selection paradigms

Feature selection is an interesting alternative to feature transform techniques such as PCA as the latter present the inconvenience of requiring that all candidate features be extracted at the test stage (before the transform found during training is applied to them). Moreover, PCA does not guarantee that noisy features will be eliminated (since noisy features may exhibit high variance) and the transformed features are difficult to interpret, which is a major drawback if one expects to gain some understanding of the qualities that best describe the classes.

By feature selection (FS), a subset of K' features is selected from a larger set of K candidates with the aim to achieve the lowest classification loss. The task is quite complex: not only is it impracticable to perform the exhaustive subset search because of the extremely high combinatorics involved, as the size of search space is 2^K when K' is not given in advance, but also it is costly to evaluate the classification loss for each candidate feature subset. Therefore feature selection is generally solved in a sub-optimal manner, usually by introducing two main simplifications:

- Brute-force search is avoided by recurring to a near-optimal search strategy.
- Instead of using the classification loss, a simpler feature selection criterion is preferred, which exploits the initial set of features intrinsically, as part of preprocessing stage, before the learning the classifiers (using only selected features). This is referred to as *filter approaches* (Sect. 5.3), as opposed to the *embedded approaches* (Sect. 5.4), where the selection is integrated in the classifier learning process.

5.3 Filter approaches

Such approaches rely on some *subset-search method* [54] and *selection criteria*—often heuristic ones, related to class separability (possibly described using a Fisher discriminant), or a measure of the association between features and classes (e.g., mutual information between them).

As for the subset search method, various strategies can be considered [54] which entail choosing a feature subset generation procedure, generally in a sequential way (e.g., forward/backward generation, sequential floating search, random generation. . .), as well as a sub-optimal search strategy, which may be either deterministic, using heuristics in the choice of the search path (e.g., adding a new feature at a time in a forward generation process), or stochastic (e.g., using simulated annealing or genetic algorithms).

A simpler yet popular approach, reduces the task to one of ranking each feature. Here, each individual feature is first scored—independently from the others—using some criterion (say a separability criterion for example). Then the features are sorted with respect to their scores and the K' top-ranked elements are retained for the classification. Such an approach is clearly sub-optimal compared with the previous search strategies, which does not prevent it from yielding satisfactory performance in practice. Its main advantage is naturally its low complexity.

5.4 Embedded feature selection

The embedded methods have attracted most of the attention in recent years, taking different forms. Hereafter, we briefly cover the most salient of such approaches.

5.4.1 Feature selection by sparsity-inducing norms

In linear-models, including support vector machines (SVM) and generalised linear models [9], feature selection is achieved using a form of regularisation, usually ℓ_1-norm regularisation in order to promote sparsity of the linear weight vector, as done in the *LASSO* [88]. The classification model estimation problem then takes the general form:

$$\min_{\boldsymbol{\beta}\in\mathbb{R}^K} \frac{1}{T}\sum_{t=1}^{T} \ell\left(y_t, \boldsymbol{\beta}^{\mathsf{T}}\mathbf{h}_t\right) + \alpha\Omega\left(\boldsymbol{\beta}\right) ; \tag{17}$$

where y_f is the class label associated to feature vector observation $\mathbf{h}_t$, $\ell(.,.)$ is a classification loss function, and $\Omega\left(\boldsymbol{\beta}\right)$ is a *sparsity-inducing norm*. This norm may be constructed in such a way to account for prior knowledge on the structure of the data, especially to perform *feature-group* selection, as opposed to feature-coefficient selection [45, 3]. Such a selection process (aka *feature-subset selection*) may be more advantageous, since it may be known in advance that some variables do not make sense when isolated from a "natural" group to which they belong. Moreover, this may allow for implicitly selecting only a subset of channels, in multi-channel setups (again provided that different feature groups are associated to them) which in practice is very valuable, as this could result in a simplification of the hardware used for capturing the data.

5.4.2 Multiple kernel learning

A set of advanced feature selection techniques have been developed for kernel-based methods [84], especially SVM classifiers, within the framework of *multiple kernel learning* (MKL) [50, 85, 78]. Here the main principle is to learn the kernel κ_0 to be used by the classifier as a convex combination of pre-defined base kernels κ_r according to: $\kappa_0(\mathbf{h},\mathbf{h}') = \sum_{r=1}^{R} \mu_r \kappa_r(\mathbf{h},\mathbf{h}')$. Now by defining the different base kernels on different feature groups (possibly different feature coefficients in the extreme case), and with a proper formulation of the classifier learning problem, involving sparsity-promoting penalties [78], only a subset of the considered kernels will have non-zero weights in the final solution, hence only a subset of features will be retained.

5.4.3 Feature selection in tree-based classifiers

In classification schemes based on trees, possibly under *boosting* or *random-forest* settings [38, chap. 10], feature selection often comes as a by-product of the classifier learning process, which may occur at various levels: either at the stage of the actual tree growing process, where at each node a particular feature (or feature set) is naturally selected; or at the level of the *ensemble* meta-classifier, which through the selection of the weak classifiers (in boosting schemes) or the random sub-sampling of the variables (in random forests), retains at the end of the learning only the subset of the most useful features. Additionally, further dimensionality reduction can be accomplished as part of a post-processing stage where efficient procedures for *variable importance determination* and *pruning* exist [38, chap. 10].

6 Temporal integration and pooling

Most of the features described above (see Sect. 3) capture specific properties of the given signal over short-time signal analysis windows (or *frames*) over which the signal can be considered stationary. Then, it is commonly assumed that the successive observations of features in different frames are statistically independent, which means that the time evolution of these features is neglected for classification. In this section, we describe several strategies, often termed *temporal integration*, to take into account the information conveyed in the temporal evolution of the signal.

6.1 Temporal integration by simple statistics

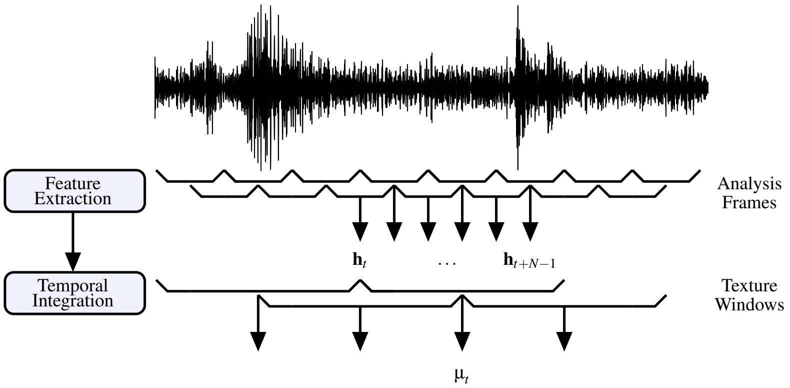


Fig. 5: Illustration of the different windows used (analysis frame and texture window

Temporal integration can be directly performed on the "instantaneous" features computed locally over short analysis frames. This so-called early integration is then commonly done over larger time windows called *texture windows* (see Fig. 5). The early temporal integration process can be represented by a function g which is applied on a sequence of feature vectors, noted $\mathbf{h}_t = [h_{1,t}\ h_{2,t}\ \ldots\ h_{K,t}]$ where $h_{f,t}$ corresponds to the f^{th} scalar feature observed in the t^{th} frame.

The aims of the integration function is to either capture short time statistics (such as the mean and covariance described below) or to more complex temporal integration using some kind of models (see Sect. 6.2). A straightforward mean for early integration is to compute first order statistics of the feature process. The *mean* integration function is then defined as

$$g_{\text{mean}}\left(\mathbf{h}_t, \ldots, \mathbf{h}_{t+N-1}\right) = \mu_t = \frac{1}{N} \sum_{k=t}^{t+N-1} \mathbf{h}_k \,. \tag{18}$$

This simple approach can be extended max-abs pooling (Chap. 5, Sect. 4.3) or to higher order statistics using for example the full co-variance matrix (or only the empirical variance of the features), the skewness or kurtosis (see for example [46, 61, 89] for some examples on music signal processing applications).

6.2 Model based integration

More sophisticated models can also be used to model the temporal dependency between successive features. It is for example possible to model the sequence of features as an autoregressive process. Such a model will capture some global spectral properties, where the level of details depend on the order of the AR model. Following the multivariate autoregressive model used in Meng [63] for music genre classification, the corresponding integration function g_{MAR} can be written as :

$$g_{\mathrm{MAR}}\left(\mathbf{h}_t, \ldots, \mathbf{h}_{t+N-1}\right) = \left[\hat{\mathbf{w}}\ \hat{\mathbf{A}}_1 \ldots \hat{\mathbf{A}}_p\right] , \tag{19}$$

where $\hat{\mathbf{w}}$ and $\{\hat{\mathbf{A}}_p\}_{p=1,\ldots,P}$ are the least-square estimators of the model parameters for the t texture window and where the p^{th} order model, denoted by $MAR(p)$ is defined as:

$$\mathbf{h}_t = \hat{\mathbf{w}} + \sum_{p=1}^{P} \mathbf{h}_{t-p}\hat{\mathbf{A}}_p + \boldsymbol{\varepsilon}_t , \tag{20}$$

with $\boldsymbol{\varepsilon}_t$ being a D-dimensional white noise vector.

A number of variations of this model have been proposed including for example the diagonal autoregressive model or the centred autoregressive model.

Direct extensions of the previous concepts aim at computing spectral characteristics of the feature sequence. Such integrated features include for example, the modulation energy of "instantaneous" MFCC features [62], the spectral moments of the feature sequence over a texture window, the STFT coefficients (or as more recently proposed the coefficients of the scaterring transform) for every feature over a texture window.

An alternative strategy will consist in incorporating some learning or classification paradigms in the feature calculation. It is for example possible to estimate the probability density of the feature sequence over a texture window and to model it using a Gaussian mixture model (GMM), GMM super-vectors or even I-vectors [27]. Since these integration approaches can be considered as part of the classification algorithm, they are not further discuss herein.

6.3 Discussion

In sound scene and event analysis, the importance accorded to temporal integration of features largely depends on the target problem. First, for a task like sound event detection, where precise estimation of event onset times is required, the use of temporal integration is rather uncommon. Instead, the temporal information of the sound scene is modelled during the classification stage by using technique such as hidden Markov models [30, 66], RNN [1, 70] or CNN for finite context [75].

The importance of temporal integration is particularly clear for other tasks like sound scene and event classification, where the decision is taken on longer segments of sound. Because of the frame-based nature of many of these features, a particular focus on temporal integration is required in order to model the distribution of the features across the full duration of the sound examples. In that case, the most common approaches are either, to classify the frame-based features before performing voting strategies, or to directly classify statistics of frame-based features computed over the full duration of the sound (see also late and early fusion techniques in Chap. 5). In the latter case, the most common way of modelling the temporal information is either to extend the feature set with their first and second order derivatives or to compute their average over time, possibly combined with more complex statistical functions [34, 49]. The use of *Recursive Quantitative Analysis* [81] on frame-based features has also proven to be effective for modelling temporal information.

7 Relation to work on speech and music processing

Speech and music are specific examples of sound signals and, as such, share many acoustical characteristics with sound scenes and sound events recordings. Speech processing is a well established field with a long history of research. Numerous features have then been proposed to characterise speech signals and used in several major classification tasks such as speech recognition or speaker identification. Music signal processing, although more recent than speech, is nevertheless another major domain of audio signal processing with a strong history.

It is therefore not surprising that a large body of features formerly introduced in speech and music research has been directly applied to sound scene or sound event recognition. ZCR, filterbanks, cepstral features, and a number of perceptually motivated features [80] were indeed proposed previously for varied sound classification tasks.

In particular, the MFCC described in Sect. 3.3, remain, even today, one of the most widely used features in sound classification since its initial use for a music processing task by Logan [55]. This is surprising since MFCC were initially designed for processing speech signals and in particular for speech recognition [23]. In fact, MFCC integrate some perception properties and, with reference to the classic speech source-filter production model, mainly discard the source part making the MFCC rather pitch independent. A direct application of MFCC for music and environmental sound analysis is surprising since 1) the pitch range is much wider in general sound signals than in speech; 2) For high pitches the deconvolution property of MFCCs does not hold anymore (e.g. MFCC become pitch dependent); and 3) MFCC are not highly correlated with the perceptual dimensions of polyphonic timbre in music signals despite their widespread use as predictors of perceived similarity of timbre [2, 66, 80]. It seems however that the MFCC‘s capacity to capture global spectral envelope properties is the main reason of their success in sound classification tasks.

However, it is worth emphasising that some recent proposals targeted features especially designed for sound scenes or sound events recognition. These include for example the matching pursuit based features proposed in Chu et al. [17], the image-based histogram features proposed in Rakotomamonjy et al. [79] or the learned matrix factorisation features [11]. Indeed, the problem of sound scene and sound event recognition is different and calls for features that are adapted to the specificities of the problem, to the scarcity of training (annotated) data and to the fact that individual classes (especially events) may be only observed in mixtures.

8 Conclusion and future directions

In this chapter we have presented an overview of the different blocks of a standard feature extraction process. The analysis of sound scene and events is a relatively new field of research in the context of sound signal analysis in general. Thus, the majority of the techniques presented in this chapter were introduced for other applications and have only later been applied to address sound scene analysis problems. In fact, many early works focused on comparing the effectiveness of different previously existing feature extraction techniques with strong inspirations from speech and music processing techniques.

We have shown that the first step in most feature extraction techniques is the choice of a suited time-frequency representation. Being at the beginning of the processing chain they play crucial role in building a sound scene analysis system. However, the choice of the representation and its parameters is rarely justified apart from stating the perceptually motivated aspect of most of them. As mentioned, many systems directly input such representations into the classification stage especially for deep learning techniques. Therefore, the performance of such systems can be limited to the quality of the representation/features used for training. Hence, the sound scene and event analysis field would benefit more in depth studies of the advantages and drawbacks of certain representation to accurately describe and discriminate the useful information in sound scenes. Moreover, new alternative representations have emerged, mostly based on scattering transforms, and have provided significant increases in performance for some problems.

We have also presented a selection of the most frequently used hand-crafted features. It is still common to see the introduction of new features for sound scene and event analysis mainly inspired from speech, music or image processing processing. The study of hand-crafted features often brings interesting insight on the content and behaviour of sound scenes. However, they are often limited to describing only specific aspects of the time-frequency information. Multiple studies have exhibited this limitation of hand crafted features by showing that combining a large variety of different features is often required to improve performance over features taken in isolation.

Finally, the most recent performance breakthroughs in sound scene and event analysis have been attained by using feature learning based on MF or deep neural network techniques. These have the advantage of automatically learning the relevant information in the data often directly from time-frequency representations. Therefore they allow for bypassing the exploration and engineering effort of choosing suited features for the task. However, deep learning techniques require their own kind of engineering effort for finding the appropriate architecture for the target task, which is highly dependent on the content and size of the datasets. In contrary, MF techniques for feature learning demands a lot less tuning effort and have shown on many occasions to be competitive with deep learning systems even when using simple classifiers. We believe that future progress in the field will be highly conditioned on the release of new larger datasets, which will further increase the effectiveness of deep learning techniques, as well as future developments in unsupervised or supervised feature learning techniques such as matrix factorisation.

References

1. Adavanne, S., Parascandolo, G., Pertila, P., Heittola, T., Virtanen, T.: Sound event detection in multichannel audio using spatial and harmonic features. In: Proc Workshop Detect Classif Acoust Scenes Events, pp. 6–10 (2016)
2. Alluri, V., Toiviainen, P.: Exploring perceptual and acoustical correlates of polyphonic timbre. Music Percept **27**(3), 223–241 (2010)
3. Bach, F.R., Jenatton, R., Mairal, J., Obozinski, G.: Optimization with Sparsity-Inducing Penalties. Tech. rep., INRIA - SIERRA Project Team (2011)
4. Battaglino, D., Lepauloux, L., Pilati, L., Evansi, N.: Acoustic context recognition using local binary pattern codebooks. Proc IEEE Workshop Appl Signal Process Audio Acoust pp. 1–5 (2015)
5. Benetos, E., Lagrange, M., Dixon, S.: Characterisation of acoustic scenes using a temporally constrained shift-invariant model. In: Proc Digit Audio Eff (2012)
6. Benetos, E., Lagrange, M., Plumbley, M.D., et al.: Detection of overlapping acoustic events using a temporally-constrained probabilistic model. In: Proc IEEE Int Conf Acoust Speech Signal Process, pp. 6450–6454. IEEE (2016)
7. Bengio, Y.: Learning deep architectures for AI. Found and Trends Mach Learn **2**(1), 1–127 (2009)
8. Bengio, Y., Courville, A., Vincent, P.: Representation learning: A review and new perspectives. IEEE Trans Pattern Anal Mach Intell **35**(8), 1798–1828 (2013)
9. Bishop, C.M.: Pattern recognition and machine learning. Springer (2006)
10. Bisot, V., Essid, S., Richard, G.: HOG and subband power distribution image features for acoustic scene classification. In: Proc Eur Signal Process Conf, pp. 719–723 (2015)
11. Bisot, V., Serizel, R., Essid, S., Richard, G.: Acoustic scene classification with matrix factorization for unsupervised feature learning. In: Proc IEEE Int Conf Acoust Speech Signal Process, pp. 6445–6449 (2016)
12. Bisot, V., Serizel, R., Essid, S., Richard, G.: Feature Learning with Matrix Factorization Applied to Acoustic Scene Classification (2016). URL https://hal.archives-ouvertes.fr/hal-01362864. HAL: working paper or preprint (hal-01362864)
13. Brown, J.C.: Calculation of a constant Q spectral transform. J Acoust Soc Am **89**(1), 425–434 (1991)
14. Burt, P., Adelson, E.: The Laplacian pyramid as a compact image code. IEEE Trans Commun **31**(4), 532–540 (1983)
15. Burt, P.J.: Fast filter transform for image processing. Comput Graphics Image Process **16**(1), 20 – 51 (1981)
16. Cauchi, B.: Non-negative matrix factorization applied to auditory scene classification. Master's thesis, ATIAM (UPMC / IRCAM / TELECOM ParisTech) (2011)

17. Chu, S., Narayanan, S., Kuo, C.C.J.: Environmental sound recognition with time-frequency audio features. IEEE Trans Audio Speech Lang Process **17**(6), 1142–1158 (2009)
18. Cichocki, A., Lee, H., Kim, Y.D., Choi, S.: Non-negative matrix factorization with α-divergence. Pattern Recognit Lett **29**(9), 1433–1440 (2008)
19. Cichocki, A., Zdunek, R., Amari, S.: Csiszar's divergences for non-negative matrix factorization: family of new algorithms. In: Proc Int Conf Indep Compon Anal Blind Signal Sep, pp. 32–39. Charleston SC, USA (2006)
20. Comon, P., Jutten, C.: Handbook of blind source separation, independent component analysis and applications. Academic Press, Elsevier (2010)
21. Cotton, C.V., Ellis, D.: Spectral vs. spectro-temporal features for acoustic event detection. In: Proc IEEE Workshop Appl Signal Process Audio Acoust, pp. 69–72. IEEE (2011)
22. Croisier, A., Esteban, D., Galand, C.: Perfect channel splitting by use of interpolation/decimation/tree decomposition techniques. In: Int Conf Inf Sci Syst, vol. 2, pp. 443–446. Patras, Greece (1976)
23. Davis, S., Mermelstein, P.: Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. IEEE Trans Acoust Speech Signal Process **28**(4), 357–366 (1980)
24. Dennis, J., Tran, H.D., Chng, E.S.: Image feature representation of the subband power distribution for robust sound event classification. IEEE Trans Audio Speech Lang Process **21**(2), 367–377 (2013)
25. Dhillon, I., Sra, S.: Generalized nonnegative matrix approximations with Bregman divergences. In: Proc Adv Neural Inf Process Syst, vol. 19, pp. 290, 283 (2005)
26. Eggert, J., Korner, E.: Sparse coding and NMF. In: Proc Int Jt Conf Neural Netw, vol. 4, pp. 2529–2533. IEEE (2004)
27. Eghbal-Zadeh, H., Lehner, B., Dorfer, M., Widmer, G.: CP-JKU submissions for DCASE-2016: a hybrid approach using binaural i-vectors and deep convolutional neural networks. Tech. rep., DCASE2016 Challenge (2016)
28. Eguchi, S., Kano, Y.: Robustifying maximum likelihood estimation. Tech. rep., Institute of Statistical Mathematics (2001)
29. Ephraim, Y., Malah, D.: Speech enhancement using a minimum mean-square error log-spectral amplitude estimator. IEEE Trans Acoust Speech Signal Process **33**(2), 443–445 (1985)
30. Eronen, A.J., Peltonen, V.T., Tuomi, J.T., Klapuri, A.P., Fagerlund, S., Sorsa, T., Lorho, G., Huopaniemi, J.: Audio-based context recognition. IEEE Trans Audio Speech Lang Process **14**(1), 321–329 (2006)
31. Fant, G.: Analysis and synthesis of speech processes. In: B. Malmberg (ed.) Manual of phonetics, chap. 8, pp. 173–277. North-Holland Publishing Company Amsterdam (1968)
32. Févotte, C., Bertin, N., Durrieu, J.L.: Nonnegative matrix factorization with the Itakura-Saito divergence. with application to music analysis. Neural Comput **21**(3) (2009)
33. Fletcher, H.: Auditory patterns. Rev Mod Phys **12**(1), 47 (1940)
34. Geiger, J.T., Schuller, B., Rigoll, G.: Large-scale audio feature extraction and SVM for acoustic scene classification. In: Proc IEEE Workshop Appl Signal Process Audio Acoust (2013)
35. Glasberg, B.R., Moore, B.C.: Derivation of auditory filter shapes from notched-noise data. Hear Res **47**(1-2), 103–138 (1990)
36. Goupillaud, P., Grossmann, A., Morlet, J.: Cycle-octave and related transforms in seismic signal analysis. Geoexploration **23**(1), 85–102 (1984)
37. Grézl, F., Karafiát, M., Kontár, S., Cernocky, J.: Probabilistic and bottle-neck features for LVCSR of meetings. In: Proc IEEE Int Conf Acoust Speech Signal Process (ICASSP), vol. 4, pp. IV–757. IEEE (2007)
38. Hastie, T., Tibshirani, R., Friedman, J.H.: The elements of statistical learning: data mining, inference, and prediction. Springer (2009)
39. Haykin, S.: Adaptive Filter Theory, 5 edn. Pearson Education (2014)
40. Henaff, M., Jarrett, K., Kavukcuoglu, K., LeCun, Y.: Unsupervised learning of sparse features for scalable audio classification. In: Proc Int Soc Music Inf Retr, vol. 11, p. 2011 (2011)

41. Hermansky, H., Ellis, D.P., Sharma, S.: Tandem connectionist feature extraction for conventional HMM systems. In: Proc IEEE Int Conf Acoust Speech Signal Process, vol. 3, pp. 1635–1638. IEEE (2000)
42. Hinton, G.E., Osindero, S., Teh, Y.W.: A fast learning algorithm for deep belief nets. Neural Comput **18**(7), 1527–1554 (2006)
43. Hotelling, H.: Analysis of a complex of statistical variables into principal components. J Educ Psychol (1933)
44. Hoyer, P.O.: Non-negative matrix factorization with sparseness constraints. J Mach Learn Res **5**, 1457–1469 (2004)
45. Jenatton, R., Audibert, J.Y., Bach, F.: Structured variable selection with sparsity-inducing norms. J Mach Learn Res **12**, 2777–2824 (2009). DOI arXiv:0904.3523
46. Joder, C., Essid, S., Richard, G.: Temporal integration for audio classification with application to musical instrument classification. IEEE Trans Audio Speech Lang Process **17**(1), 174–186 (2009)
47. Kobayashi, T., Ye, J.: Acoustic feature extraction by statistics based local binary pattern for environmental sound classification. In: Proc IEEE Int Conf Acoust Speech Signal Process, pp. 3052–3056. IEEE (2014)
48. Komatsu, T., Toizumi, T., Kondo, R., Senda, Y.: Acoustic event detection method using semi-supervised non-negative matrix factorization with mixtures of local dictionaries. In: Proc IEEE AASP Chall Detect Classif Acoust Scenes Events, pp. 45–49 (2016)
49. Krijnders, J., Holt, G.A.T.: A tone-fit feature representation for scene classification. Proc IEEE AASP Chall Detect Classif Acoust Scenes Events (2013)
50. Lanckriet, G.R.G., Bartlett, P., Jordan, M.I.: Learning the kernel matrix with semidefinite programming. J Mach Learn Res **5**, 27–72 (2004)
51. Lee, D.D., Seung, H.S.: Algorithms for non-negative matrix factorization. In: Proc Adv Neural Inf Process Syst, pp. 556–562 (2001)
52. Lee, L., Rose, R.C.: Speaker normalization using efficient frequency warping procedures. In: Proc IEEE Int Conf Acoust Speech Signal Process, vol. 1, pp. 353–356. IEEE (1996)
53. Lee, L., Seung, S.: Learning the parts of objects with nonnegative matrix factorization. Nat **401**, 788–791 (1999)
54. Liu, H., Motoda, H.: Feature selection for knowledge discovery and data mining, 2nd edn. Kluwer academic publishers (2000)
55. Logan, B.: Mel frequency cepstral coefficients for music modeling. In: Proc Int Soc Music Inf Retr Conf (2000)
56. Lostanlen, V., Andn, J.: Binaural scene classification with wavelet scattering. Tech. rep., DCASE2016 Challenge (2016)
57. Mairal, J., Bach, F., Ponce, J., Sapiro, G.: Online dictionary learning for sparse coding. In: Proc Int Conf Mach Learn, pp. 689–696 (2009)
58. Mallat, S.: A wavelet tour of signal processing. Academic press (1999)
59. Mallat, S.: Group invariant scattering. Commun Pure Appl Math **65**(10), 1331–1398 (2012)
60. Mallat, S.G.: Multifrequency channel decompositions of images and wavelet models. IEEE Trans Acoust Speech Signal Process **37**(12), 2091–2110 (1989)
61. Mandel, M., Ellis, D.: Song-level features and SVMs for music classification. In: Proc Int Soc Music Inf Retr Conf (2005)
62. McKinney, M.F., Breebart, J.: Features for audio and music classification. In: Int Symp Music Inf Retr, pp. 151–158 (2003)
63. Meng, A.: Temporal feature integration for music organisation. Ph.D. thesis, Technical University of Denmark (2006)
64. Mesaros, A., Heittola, T., Dikmen, O., Virtanen, T.: Sound event detection in real life recordings using coupled matrix factorization of spectral representations and class activity annotations. In: Proc IEEE Int Conf Acoust Speech Signal Process, pp. 151–155 (2015)
65. Mesaros, A., Heittola, T., Virtanen, T.: TUT database for acoustic scene classification and sound event detection. In: Proc Eur Signal Process Conf (2016)
66. Mesaros, A., Virtanen, T.: Automatic recognition of lyrics in singing. EURASIP J Audio Speech Music Process **2010**(1), 546,047 (2010)

67. Murphy, K.P.: Machine learning: a probabilistic perspective (2012)
68. O'Shaughnessy, D.: Speech communication: human and machine. Addison-Wesley series in electrical engineering. Addison-Wesley Pub. Co. (1987)
69. O'Shaughnessy, D.: Linear predictive coding. IEEE Potentials **7**(1), 29–32 (1988)
70. Parascandolo, G., Huttunen, H., Virtanen, T.: Recurrent neural networks for polyphonic sound event detection in real life recordings. In: Proc IEEE Int Conf Acoust Speech Signal Process, pp. 6440–6444. IEEE (2016)
71. Patterson, R.D., Robinson, K., Holdsworth, J., McKeown, D., Zhang, C., Allerhand, M.: Complex sounds and auditory images. In: Proc Int Symp Hear, vol. 83, pp. 429–446. Oxford, UK: Pergamon (1992)
72. Peeters, G.: A large set of audio features for sound description (similarity and classification) in the CUIDADO project. Tech. rep., IRCAM, Paris, France (2004)
73. Peltonen, V., Tuomi, J., Klapuri, A., Huopaniemi, J., Sorsa, T.: Computational auditory scene recognition. In: Proc Int Conf Acoust Speech Signal Process, vol. 2, pp. II–1941 (2002)
74. Phan, H., Hertel, L., Maass, M., Koch, P., Mertins, A.: Car-forest: Joint classification-regression decision forests for overlapping audio event detection (DCASE). Tech. rep., DCASE2016 Challenge (2016)
75. Piczak, K.J.: Environmental sound classification with convolutional neural networks. In: Proc Int Workshop Mach Learn Signal Process, pp. 1–6. IEEE (2015)
76. Plinge, A., Grzeszick, R., Fink, G.A.: A bag-of-features approach to acoustic event detection. In: Proc IEEE Int Conf Acoust Speech Signal Process, pp. 3704–3708. IEEE (2014)
77. Rakotomamonjy, A.: Enriched supervised feature learning for acoustic scene classification. Tech. rep., DCASE2016 Challenge (2016)
78. Rakotomamonjy, A., Bach, F.R., Canu, S., Grandvalet, Y.: SimpleMKL. J Mach Learn Res **9**, 2491–2521 (2008)
79. Rakotomamonjy, A., Gasso, G.: Histogram of gradients of time-frequency representations for audio scene classification. IEEE/ACM Trans Audio Speech Lang Process **23**(1), 142–153 (2015)
80. Richard, G., Sundaram, S., Narayanan, S.: An overview on perceptually motivated audio indexing and classification. Proceedings of the IEEE **101**(9), 1939–1954 (2013)
81. Roma, G., Nogueira, W., Herrera, P.: Recurrence quantification analysis features for environmental sound recognition. In: Proc IEEE Workshop Appl Signal Process Audio Acoust (2013)
82. Salamon, J., Bello, J.B.: Feature learning with deep scattering for urban sound analysis. In: Proc Eur Signal Process Conf, pp. 724–728. IEEE (2015)
83. Salamon, J., Bello, J.P.: Unsupervised feature learning for urban sound classification. In: Proc IEEE Int Conf Acoust Speech Signal Process, pp. 171–175 (2015)
84. Shölkopf, B., Smola, A.J.: Learning with kernels: support vector machines, regularization, optimization, and beyond. MIT Press, Cambridge, MA, USA (2002)
85. Sonnenburg, S., Rätsch, G., Schäfer, C., Schölkopf, B.: Large scale multiple kernel learning. J Mach Learn Res **7**, 1531–1565 (2006)
86. Stevens, S.S., Volkmann, J., Newman, E.B.: A scale for the measurement of the psychological magnitude pitch. J Acoust Soc Am **8**(3), 185–190 (1937)
87. Sun, D., Mazumder, R.: Non-negative matrix completion for bandwidth extension: A convex optimization approach. Proc IEEE Int Workshop Mach Learn Signal Process (2013)
88. Tibshirani, R.: Regression shrinkage and selection via the lasso. J R Stat Soc Series B Stat Methodol pp. 267–288 (1996)
89. Tzanetakis, G., Cook, P.: Musical genre classification of audio signals. In: IEEE Trans Speech Audio Process (2002)
90. Valero, X., Alías, F.: Gammatone cepstral coefficients: biologically inspired features for non-speech audio classification. IEEE Trans Multimed **14**(6), 1684–1689 (2012)
91. Vincent, P., Larochelle, H., Bengio, Y., Manzagol, P.A.: Extracting and composing robust features with denoising autoencoders. In: Proc Int Conf Mach Learn, pp. 1096–1103. ACM (2008)

92. Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., Manzagol, P.A.: Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. J Mach Learn Res **11**, 3371–3408 (2010)
93. Widrow, B., Stearns, S.D.: Adaptive signal processing, 1 edn. Englewood Cliffs, NJ, Prentice-Hall, Inc., 1985, 491 p. (1985)
94. Ye, J., Kobayashi, T., Murakawa, M., Higuchi, T.: Acoustic scene classification based on sound textures and events. In: Proc Annu Conf Multimed, pp. 1291–1294 (2015)
95. Zwicker, E., Terhardt, E.: Analytical expressions for critical-band rate and critical bandwidth as a function of frequency. J Acoust Soc Am **68**(5), 1523–1525 (1980)